Texture Segregation in Chromatic Element-Arrangement Patterns

Jacob Beck
Department of Cognitive and Neural Systems
Boston University
Boston, MA 02215

Abstract

An experiment compared the perceived segregation of element-arrangement patterns composed of the same two element types arranged in vertical stripes in the top and bottom regions and in a checkerboard pattern in the middle region. The elements in the patterns differed in hue. Patterns were equated for either luminance or for brightness. The experiment investigated the effects of the hues of the squares, pattern size, and background luminance on the segregation of the element-arrangement patterns. Perceived segregation was strong with a low luminance black background but was greatly decreased by a high luminance white background. Perceived segregation on a black background was stronger for blue and green patterns than for green and yellow patterns. Perceived segregation increased with a decrease in pattern size. Hue similarity, as rated by subjects in a separate procedure, was a relatively weak factor for predicting perceived segregation. The results are consistent with the hypothesis that perceived segregation is a function of cone contrasts.

1: Introduction

An element-arrangement pattern is composed of two types of elements that differ in the ways in which they are arranged in different regions of the pattern. Figure 1 illustrates an element-arrangement pattern in which the elements are red and blue squares arranged in a striped pattern in the top and bottom regions and in a checkerboard pattern in the middle region. The pattern is shown with black and white interspaces. Perceived segregation is strong with black interspaces but is diminished by white interspaces. A brief review of past research is presented to provide context for the experiment reported.

0-8186-7644-2 $5.00

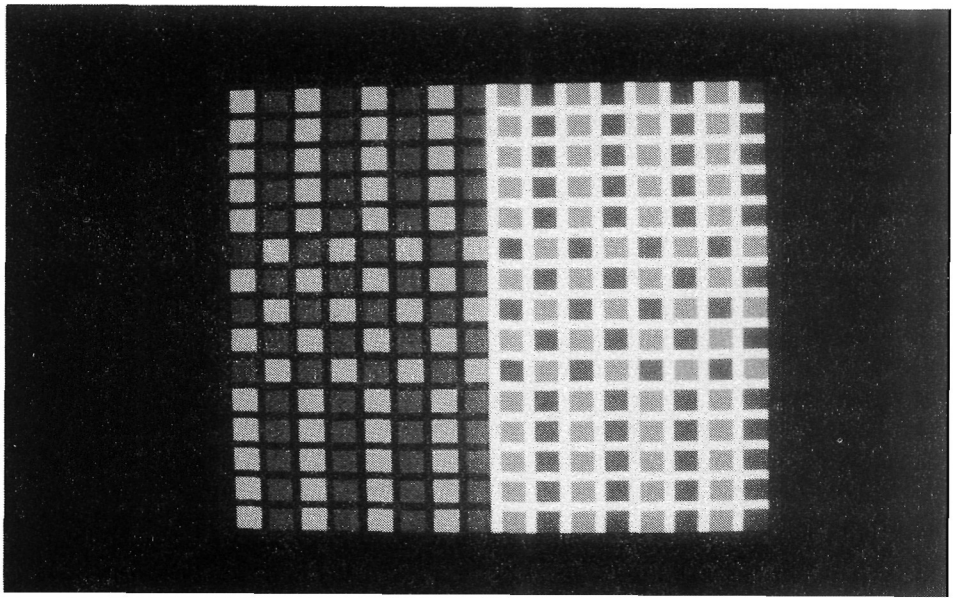


Figure 1. An element-arrangement pattern composed of red and blue squares with black and white interspaces. Perceived segregation with black interspaces is strong while perceived segregation with white interspaces is reduced. Unfortunately the photographic reproduction process distorts the image and the effect may be diminished.

1.1: Achromatic Element-Arrangement Patterns

Research with achromatic element-arrangement patterns indicates that the information for texture segregation takes place at a level of representation preceding the specification of the squares and their properties. First, texture segregation in element-arrangement patterns is not a direct function of the lightness differences of the squares. A striking finding reported in [1] was that in element-arrangement patterns of light and dark squares a large lightness difference could fail to yield strong texture segregation while a smaller lightness difference could yield strong segregation. Second, texture segregation is not impaired by contour misalignment. Judgments of perceived segregation were the same when the elements composing an element-arrangement pattern were aligned squares, misaligned squares, circles or blobs [2]. Third, texture segregation in an element-arrangement pattern fails to scale. Sutter, Beck and Graham [3] found that perceived segregation was strongest for patterns whose fundamental spatial frequency (the distance between the centers of two columns of the same type of square) was approximately 4 cycles per degree. Patterns with higher or lower fundamental spatial-frequencies segregated less strongly.

Beck, Sutter, and Ivry [4] showed that the perceived segregation of achromatic element-arrangement patterns was qualitatively consistent with the hypothesis that differences in the outputs of spatial frequency channels underlie the perceived segregation. They proposed that the differential responses of oriented simple cell-like mechanisms to the striped and checked regions of an element-arrangement pattern is the basis for the perceived segregation. For achromatic element-arrangement patterns, Sutter, Beck, and Graham [3] showed that the receptive fields that show strikingly different outputs to the different arrangements of

the squares in the striped and checked regions are the large receptive fields that are sensitive to the fundamental spatial frequency of the texture pattern. These receptive fields match the period of the pattern and signal the differences in the overall pattern of squares in the striped and checked regions. In the striped region the changes of overall luminance occur in the horizontal direction, strongly stimulating vertically oriented receptive fields, and in the checkerboard region changes of overall luminance occur in a direction 45 degrees from horizontal, strongly stimulating obliquely oriented receptive fields (see Figure 2). They proposed that the differences in the outputs of these receptive fields are used by the visual system to establish boundaries separating the regions of the pattern.

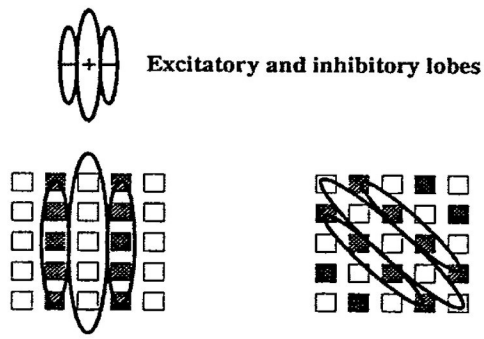


Figure 2. An illustration of how the responses of large oriented receptive fields sensitive to the fundamental frequency of a pattern can account for the segregation of achromatic element-arrangement patterns composed of light and dark squares. Top: Excitatory and inhibitory areas of an even symmetric receptive field. Bottom left: Large vertical receptive fields respond strongly to the vertical columns of squares in the striped region. Bottom right: Large oblique receptive fields respond strongly to the diagonal columns of squares in the checkerboard region.

The large receptive fields that match the period of a pattern and respond differentially to the striped and checkerboard arrangements of the squares do not have the right properties to signal the lightness of the squares because they average over several squares. Perceived segregation would thus not be expected to be a simple function of the lightness differences of the squares. Similarly, large receptive fields would also not be sensitive to edge alignment. Perceived segregation would therefore not be impaired by the misalignment of the squares. Perceived segregation would also not be expected to scale since perceived segregation is a function of the visual system's sensitivity to the fundamental spatial frequency. Proportionally reducing the overall size to a pattern would increase perceived segregation up to the point where the fundamental spatial frequency of the pattern has a spatial frequency at the peak of the contrast sensitivity function.

Sutter, Beck, and Graham [3] further found that the perceived segregation in an element-arrangement pattern is minimal when the area times contrast of large and small squares were equal. The area times contrast of the large and small squares is the same when the greater area of the large square is compensated for by the higher contrast of the small square. Squares that have the same area times contrast produce the same output at the fundamental frequency of the pattern, i.e., the frequency which when the excitatory region of a receptive field falls on one column of squares, the inhibitory region of the receptive field falls on the adjacent column of squares (see Figure 2). Although the contrast ratio—the ratio of the

contrasts of the two square types with the background—at which the minimum perceived segregation occurred was correctly predicted by the outputs of simple cell-like mechanisms, the amount of segregation at this minimum was incorrectly predicted. The amount of perceived segregation depended also on the difference in the sizes of the squares. When the area times contrast of the large and small squares were equated, perceived segregation was greater as the size difference between the large and small squares increased. One way of accounting for this discrepancy is by a more complicated spatial-frequency model in which the initial linear filtering is followed by a rectification and a second filtering at a lower spatial frequency [3]. Graham, Beck, and Sutter ([5]; see also [6]) showed that texture segregation in element-arrangement patterns cannot be explained in terms of solely linear operations, and the application of spatial frequency analysis to texture segregation involves at least two nonlinearities. One nonlinearity is an intensity-dependent nonlinearity which can be accounted for by either sensory adaptation occurring before the channels or by a compressive intracortical interaction among neuronal responses which normalizes the responses [7]. The second nonlinearity is a rectification-like nonlinearity that is like that presumed to occur in complex cells [8].

1.2: Chromatic Element-Arrangement Patterns

Beck [9] and Pessoa, Beck, and Mingolla [10] investigated the perceived segregation of element-arrangement patterns composed of equiluminant squares differing in hue. They found that the luminance of the interspace region (i.e., the spaces between squares) strongly affected perceived segregation, whereas the luminance of the surround (i.e. the space surrounding a pattern) affected perceived segregation to a minor degree. For element-arrangement patterns composed of squares differing in hue, perceived segregation was strongly interfered with by high luminance interspaces but not by low luminance interspaces (see Figure 1). It is important to note that the squares composing an element-arrangement pattern do not have to be at precise equal luminance for perceived segregation to be diminished by a high luminance interspaces. Pessoa, Beck, and Mingolla [10] found that perceived segregation tended to decrease with increasing luminance of the interspaces when the squares composing an element-arrangement pattern differed in luminance. Perceived segregation varied approximately inversely with the ratio of the background luminance to the higher luminance square. Pessoa, Beck, and Mingolla [10] also found that perceived segregation was approximately constant for constant ratios of interspace luminance to square luminance. Stereoscopic cues that caused the squares composing the element-arrangement pattern to be seen in front of the interspaces did not greatly improve perceived segregation with high luminance interspaces. As in the case of achromatic element-arrangement patterns, these results suggest that the explanation of the perceived segregation of chromatic element-arrangement patterns is in terms of the early visual mechanisms that encode hue.

2: Blue and Green and Green and Yellow Element-Arrangement Patterns

Scott Oddo, Ennio Mingolla and I are investigating texture segregation in chromatic element-arrangement patterns. We are studying the degree to which brightness differences, hue similarity, and differences in cone contrasts account for differences in the perceived segregation of chromatic element-arrangement patterns. I present a preliminary report of our results.

Five subjects rated the perceived segregation of element-arrangement patterns composed of equal luminance blue ($x = .146$, $y = .061$) and green ($x = .294$, $y = .551$)[1] squares and of green ($x = .295$, $y = .529$) and yellow ($x = .457$, $y = .411$)[1] squares on a rating scale from 0 to 4. The luminances of the blue square in the element-arrangement pattern of blue and green squares and of the green square in the element-arrangement pattern of green and yellow squares were set at 2.5 ft.-L. Equiluminance of the blue and green hues and of the green and yellow hues was established using the criterion of minimally distinct borders [11]. The procedure used is described in Pessoa, Beck and Mingolla [10]. Hues of equal luminance are often not of equal brightness. The subjects also rated the perceived segregation of element-arrangement patterns composed of blue and green squares and of green and yellow squares judged equal in brightness. Equal brightness values were determined for each subject by having the subject make heterochromatic brightness judgments. Subjects adjusted the brightness of the green squares in an element-arrangement pattern composed of blue and green squares to be equal in brightness to that of the blue squares, and the brightness of the yellow squares in an element-arrangement pattern composed of green and yellow squares to be equal in brightness to that of the green squares. The subjects also rated the similarity of the two hues composing an element-arrangement pattern for similarity on a scale from 0 to 4. The overall size of the patterns was varied by proportionally decreasing the sizes and the separations of the squares making up the pattern. The fundamental spatial frequency of the patterns (the period between two columns of squares of the same hue) was .5, 1, and 2 cycles per degree. The stimuli were presented on a high luminance white background (16.3 ft.-L., $x = .312$, $y = .325$) and on a low luminance black background (.23 ft.-L., $x = .248$, $y = .254$).

2.1: Results and Discussion

Figure 3 shows the mean segregation ratings with a black background and Figure 4 with a white background as a function of pattern size. Perceived segregation was significantly greater with a black background than with a white background for both the blue and green and green and yellow element-arrangement patterns ($F(1,4) = 93.4$; $p < .05$). The interactions of Background $\times$ Hue was significant ($F(1,4) = 14.3$; $p < .05$). On a black background the element-arrangement patterns composed of blue and green squares segregated significantly more strongly than the element-arrangement patterns composed of green and yellow squares ($F(1,4) = 41.7$; $p < .05$). On a white background the blue and green element-arrangement pattern failed to significantly segregate more strongly than the green and yellow element-arrangement pattern ($F(1,4) = 1.02$; $p > .05$) On both black and white backgrounds, perceived segregation significantly increased with a decrease in pattern size (an increase in the number of cycles per degree) ($F(2,8) = 46.0$; $p < .05$). The interaction of Background Luminance $\times$ Cycles per Degree was significant ($F(2,8) = 30.1$; $p < .05$). This interaction reflects the greater increase in perceived segregation with decreased pattern size on a black background than on a white background.

Brightness: Chromatic hues of equal luminance need not be of equal brightness. A blue hue is commonly seen as brighter than an equiluminant green hue while equiluminant green and yellow hues are commonly seen to be more nearly equal in brightness [12]. One possibility, therefore, is that the greater perceived segregation of the blue and green element-arrangement patterns than of the green and yellow element-arrangement patterns is due

[1]These are the mean chromaticity coordinates of the hues judged by subjects to be of equal luminance. The chromaticity coordinates for the equal brightness hues were similar.

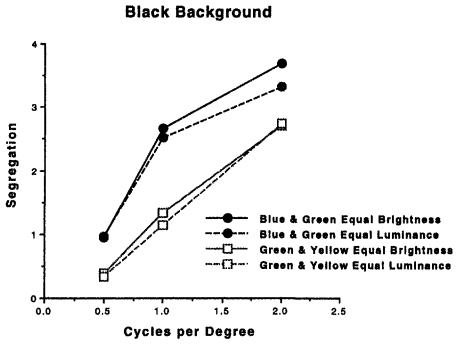


Figure 3. Mean segregation ratings on a black background plotted as a function of pattern size.

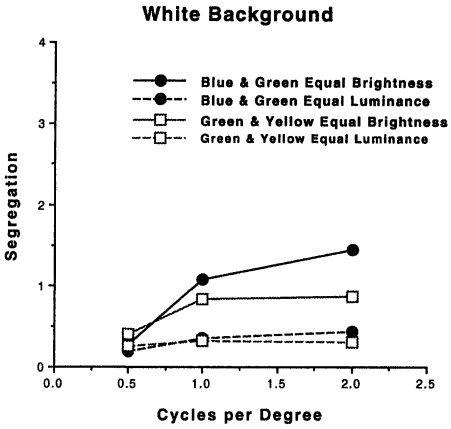


Figure 4. Mean segregation ratings on a white background plotted as a function of pattern size.

to the greater brightness difference between the equiluminant blue and green hues than between the equiluminant green and yellow hues. The ratio of the luminance of the green square to the luminance of the blue square and of the luminance of the yellow square to the luminance of the green square at equal brightness is an ordinal measure of the brightness difference between equiluminant hues. Figure 5 shows the mean ratios of the luminance of the green square to the luminance of the blue square and of the luminance of the yellow square to the luminance of the green square at equal brightness on a black background. Figure 6 shows the mean luminance ratios on a white background. The brightness difference between blue and green squares was significantly greater than between the green and yellow squares on both black and white backgrounds ($F(1,4) = 8.51; p < .05$). The brightness differences are in accord with the judgments of greater segregation for the blue and green equiluminant element-arrangement patterns than for the green and yellow equiluminant element-arrangement patterns. However, Figures 3 and 4 show that the equal brightness and equal luminance stimuli segregate alike. This clearly indicates that brightness differences are not the principal factor underlying the difference in the perceived segregation of the blue and green and the green and yellow element-arrangement patterns.

The greater luminance ratios yielding equal brightness of the hues with a white background than with a black background is the result of the inhibition of brightness by the higher luminance white background. Without knowing the degree of inhibition one can not determine whether the relative brightness differences of the hues on the black background were greater than the brightness differences on a white background. One is not able therefore to decide whether the greater segregation on a black background than on a white background reflects the greater brightness difference between the hues on a black background than on a white background. In a second experiment subjects rated the perceived segregation of element-arrangement patterns composed of equiluminant purple ($x = .310$, $y = .162$) and gray ($x = .297$, $y = .301$)[1] squares on black and white backgrounds. The purple square was set at 2.5 ft.-L. The luminances and chromaticity coordinates of the back

[1]These are the mean chromaticity coordinates of the hues judged by subjects to be of equal luminance. The chromaticity coordinates for the equal brightness hues were similar.

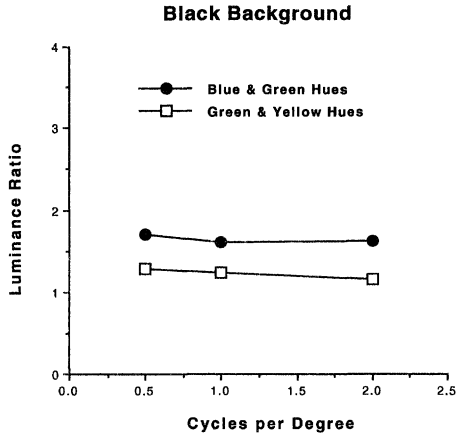


Figure 5. Mean ratios of the luminance of the green square to the luminance of the blue square and of the luminance of the yellow square to the luminance of the green square when the blue and green squares and the green and yellow squares were judged to be equally bright. The mean ratios are for a black background and are plotted as a function of pattern size.

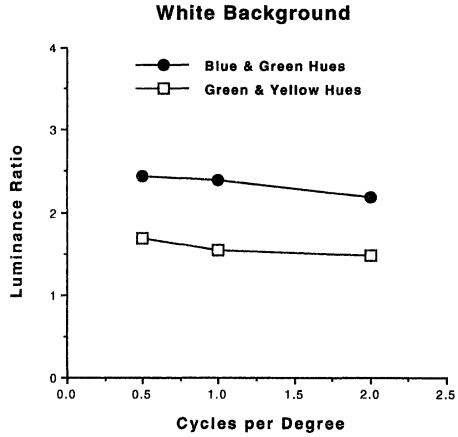


Figure 6. Mean ratios of the luminance of the green square to the luminance of the blue square and of the luminance of the yellow square to the luminance of the green square when the blue and green squares and the green and yellow squares were judged to be equally bright. The mean ratios are for a white background and are plotted as a function of pattern size.

and white backgrounds were as in the first experiment. Figure 7 shows the mean segregation ratings with black and white backgrounds as a function of pattern size. As with the blue and green and green and yellow element-arrangement patterns, perceived segregation was dramatically better on a black background than on a white background ($F(1,4) = 54.07$; $p < .05$). The gray square in an element-arrangement pattern was assigned a value of 100 and subjects also made magnitude estimations of the brightnesses of the purple squares. Figure 8 shows the magnitude estimations of brightness. The purple squares were judged to be slightly brighter than the gray squares on a black background than on a white background. However, there is not enough difference in the magnitude estimations of brightness to explain the effect of background luminance on perceived segregation.

Similarity: A second possibility is that perceived segregation is a function of hue similarity. Figure 9 shows the mean similarity ratings with a black background and Figure 10 shows the mean similarity ratings with a white background as a function of pattern size. The green and yellow hues overall were judged slightly more similar than the blue and green hues. An Anova of the similarity ratings showed that the differences in the similarity ratings of the blue and green and the green and yellow hues were not significant ($F(1,4) = 2.53$; $p > .05$). The correlations of the mean segregation judgments with the mean similarity judgments was not significant with a white background ($t(10) = 1.13$, $p > .05$). The correlation of the mean segregation judgments with the mean similarity judgments was $-.69$ with a black background ($t(10) = 3.01$, $p < .05$). However, the slope of the regression equation was -3.5 and the intercept of the regression equation was 6.6. If perceived segregation were

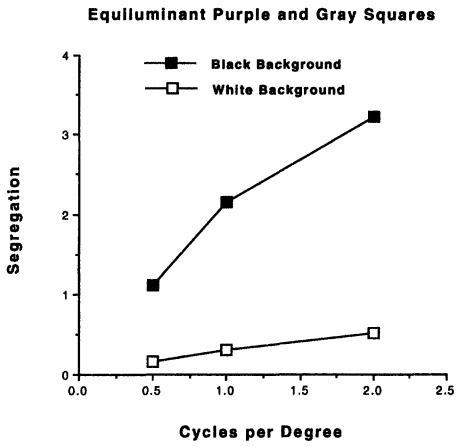


Figure 7. Mean segregation ratings on black and white backgrounds as a function of pattern size.

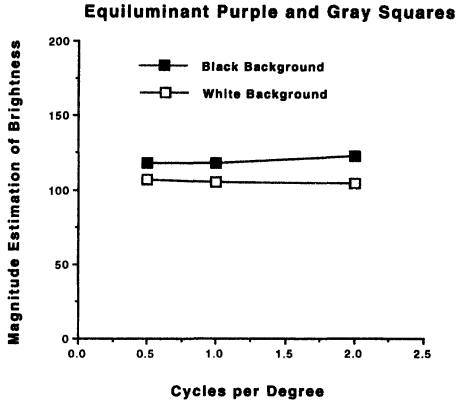


Figure 8. Mean magnitude estimations of the brightnesses of the purple squares plotted as a function of pattern size. The gray squares were assigned a value of 100.

a linear function of the similarity of the hues, the slope of the regression equation should be -1.0 and the intercept of the regression equation should be 4.0. This suggests that hue similarity is not the explanation for the greater perceived segregation of the blue and green element-arrangement patterns than of the green and yellow element-arrangement patterns.

Cone Contrasts: A third possibility is that what is important are the early visual mechanisms that encode the differences in hue of the squares composing an element-arrangement texture pattern. For the perceived segregation of an achromatic element-arrangement pattern, the luminance contrasts of the squares with the background is an important factor [1]. The analogous variable for chromatic patterns are the chromatic contrasts of the squares with the background. Cone contrasts are a measure of the hue differences encoded by the early visual mechanisms. The cone contrasts can be estimated from the long (L-cone), middle (M-cone), and short-wavelength (S-cone) cone responses. Macleod and Boynton [13] present a transformation for converting the Judd color matching functions (or tristimulus values $\bar{x}$, $\bar{y}$, and $\bar{z}$) into L-, M-, and S-cone responses. The cone contrasts are $\Delta L/L$, $\Delta M/M$ and $\Delta S/S$—the numerators are the differences between the long, middle, and short-wavelength cone responses to the squares and their responses to the background; the denominators are the cone responses to the background. Table 1 shows the mean cone contrasts for the equal luminance and equal brightness patterns on black and white backgrounds. The cone contrasts are consistent with the greater segregation of the blue and green element-arrangement patterns than of the green and yellow element-arrangement patterns. For both the equal luminance and equal brightness element-arrangement patterns, there is a large difference between the S-cone contrasts of the blue and green hues. For the green and yellow hues, which showed weaker texture segregation, there are no strikingly large differences in the contrasts of the three cones. The cone contrast responses of the equiluminant and equal brightness stimuli are also similar. The main difference is that the cone contrast differences for the L- and M-cones are greater for the equal brightness stimuli than for the equiluminant stimuli. This is to be expected since the equal brightness stimuli

differ in luminance.

Table 1. Estimated Cone Contrasts with a Black Background

	Equal Luminance Patterns			Equal Brightness Patterns		
Hue	L-cone	M-Cone	S-cone	L-cone	M-cone	S-cone
Blue[2]	6.38	11.35	54.73	6.38	11.35	54.73
Green	9.28	9.80	0.96	17.14	18.27	1.61
SE	1.02	1.07	0.17	5.40	5.79	0.52
Green[2]	9.88	10.44	0.91	9.88	10.44	0.91
Yellow	10.54	7.35	0.67	13.11	9.37	0.86
SE	0.61	0.36	0.05	1.68	1.25	0.12

Graham [6] hypothesized that the decrease in the perceived segregation of an achromatic element-arrangement pattern composed of squares differing in lightness with a high background luminance is due to a compressive intensity nonlinearity that abolishes the differences in the neural responses to the high and low luminance squares composing the pattern. Analogously, a high luminance background which strongly stimulates the L-, M-, and S-cones would be expected to decrease the differences in the cone responses to the hues of the two squares. Table 2 shows the cone contrasts for the blue and green, and green and yellow element-arrangement patterns with white backgrounds. Note that the differences in the cone contrast on a white background are not nearly as great as those with a black background. As mentioned above, Pessoa, Beck, and Mingolla [10] found that equal ratios of background luminance to hue luminance yielded approximately the same perceived segregation. It should be noted that keeping the ratio of the background luminance to the hue luminance constant leaves the cone contrasts constant. The overall results are therefore consistent with the hypothesis that perceived segregation in element-arrangement patterns is determined by the encoding of hue by early visual processes.

Table 2. Estimated Cone Contrasts with a White Background

	Equal Luminance Patterns			Equal Brightness Patterns		
Hue	L-cone	M-Cone	S-cone	L-cone	M-cone	S-cone
Blue[2]	−0.90	−0.81	−0.37	−0.90	−0.81	−0.37
Green	−0.86	−0.84	−0.95	−0.66	−0.60	−0.92
SE	0.01	0.02	0.00	0.10	0.12	0.02
Green[2]	−0.85	−0.83	−0.95	−0.85	−0.83	−0.95
Yellow	−0.84	−0.87	−0.96	−0.75	−0.80	−0.95
SE	0.01	0.01	0.00	0.11	0.09	0.01

Pattern Size: Contrast sensitivity for chromatic gratings tend to decrease beyond 1 cycle/deg [14]. The increase in perceived segregation with decreasing pattern size is therefore not ascribable to the chromatic contrast sensitivity function. It is also not explainable by any of the three factors we have studied, i.e., differences in the brightness, similarity, or

[2]The hues were set by the experimenter. SE = the standard error of the mean cone contrasts calculated from subjects' equal luminance and equal brightness judgments.

cone contrasts of the hues in an element-arrangement pattern. The magnitude estimations of brightness presented in Figure 8 were similar across the three spatial scales and do not account for the effect of pattern size. The rated similarity of the blue and green hues and of the green and yellow hues shown in Figures 9 and 10 also did not vary with pattern size, and would not explain the effects of pattern size on perceived segregation of the element-arrangement patterns. Scaling of the patterns also leaves the cone contrasts constant. One hypothesis is that the high spatial frequency information in the element-arrangement patterns interferes with perceived segregation and that this interference is reduced when the high spatial frequencies are outside the range of visual sensitivity. We are examining this possibility.

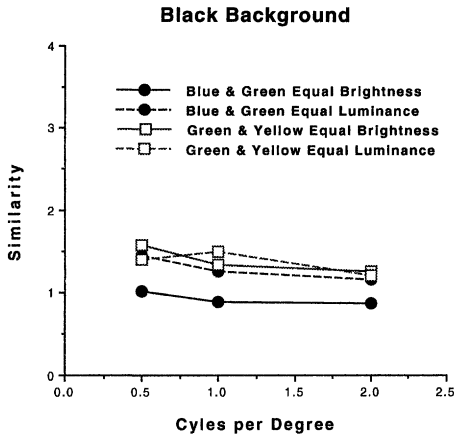


Figure 9. Mean similarity ratings on a black background plotted as a function of pattern size.

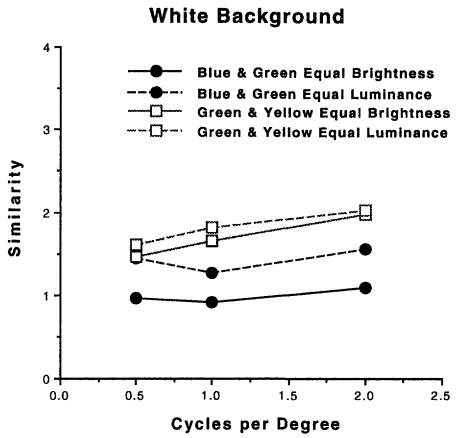


Figure 10. Mean similarity ratings on a white background plotted as a function of pattern size.

Acknowledgments

We thank Rhea Eskew for helpful discussions concerning the calculations of the cone contrasts, and Ken Nakayama for the use of the Minolta CS-100 chroma-meter. The research was supported in part by the Office of Naval Research (ONR N00014-J-4100, ONR N00014-J-4015) and the Air Force Office of Scientific Research (AFOSR F49620-92-J-0334).

References

[1] Beck, J., Graham, N., and Sutter, A. (1991) Lightness differences and the perceived segregation of regions and populations. *Perception and Psychophysics* **49**, 257–269.

[2] Beck, J. (1993) Visual processing in texture segregation. In: *Visual Search 2*, D. Brogan, A. Gale, and K. Carr (Eds.), London: Taylor and Francis (pp. 1–35).

[3] Sutter, A., Beck, J., and Graham, N. (1989) Contrast and spatial variables in texture segregation: Testing a simple spatial frequency channels model. *Perception and Psychophysics* **46**, 312–332.

[4] Beck, J., Sutter, A., and Ivry, R. (1987) Spatial frequency channels and perceptual grouping in texture segregation. *Computer Vision, Graphics, and Image Processing* **37**, 299–325.

[5] Graham, N., Beck, J., and Sutter, A. (1992) Nonlinear processes in spatial frequency channel models of perceived texture segregation: Effects of sign and amount of contrast. *Vision Research* **32**, 719–743.

[6] Graham, N. (1994) Nonlinearities in texture segregation. In: Ciba Foundation Symposium, 184, *Higher Order Processing in the Visual System*, Chichester: Wiley (pp. 309–329).

[7] Grossberg, S. and Mingolla, E. (1985) Neural dynamics of perceptual grouping: Textures, boundaries, and emergent features. *Perception and Psychophysics* **38**, 141–171.

[8] Spitzer, H. and Hochstein, S. (1985) A complex-cell receptive field model. *Journal of Neurophysiology* **53**, 1266–1286.

[9] Beck, J. (1994) Interference in the perceived segregation of equal luminance element-arrangement texture patterns. *Perception and Psychophysics* **56**, 424–430.

[10] Pessoa, L., Beck, J., and Mingolla, E. (1996) Perceived segregation in chromatic element-arrangement patterns: High intensity interference. *Vision Research* (in press).

[11] Boynton, R.M. and Kaiser P. (1968) Vision: The additivity law made to work in heterochromatic photometry with bipartite fields. *Science* **161**, 366–368.

[12] Wyszecki, G. and Stiles, W.S. (1982) *Color Science: Concepts and Methods, Quantitative Data and Formulae.* New York: Wiley.

[13] Macleod, D.I.A. and Boynton, R.M. (1979) Chromaticity diagram showing cone excitation by stimuli of equal luminance. *Journal of the Optical Society of America* **69**, 1183–1186.

[14] Mullen, K.T. (1985) The contrast sensitivity of human color vision to red-green and blue-yellow chromatic gratings. *Journal of Physiology* (London) **359**, 381–400.

Multiresolution GMRF Models for Image Segmentation*

Rama Chellappa† and Santhana Krishnamachari ‡

† Department of Electrical Engineering and
Center for Automation Research
University of Maryland
College Park, MD 20742

‡ Image Processing Department
Communication Technology Division
COMSAT Laboratories
Clarksburg, MD 20871

Abstract

A multiresolution model for Gauss Markov random fields **(GMRF)** *with application to texture segmentation is presented. Coarser resolution sample fields are obtained by subsampling the sample field at the fine resolution. Although the Markov property is lost under such resolution transformation, coarse resolution non-Markov random fields can be effectively approximated by Markov fields. We present a local conditional distribution invariance approximation to estimate the GMRF parameters at coarser resolutions from the fine resolution parameters. Our experiments with synthetic, Brodatz texture and real satellite images show that this multiresolution technique results in a better segmentation and requires lesser computation than the single resolution algorithm.*

1: Introduction

There has been an increasing emphasis on using statistical techniques for modeling and analyzing images. Typical image processing problems have the following aspects to be dealt with: the identification of an appropriate model that reflects the prior beliefs and knowledge about the family of images that are to be analyzed, the selection of a proper observation model that reflects the nature of the transformations these images undergo during observation, and the selection of a suitable error criterion to be optimized. For many image processing problems, such as image enhancement, image restoration, texture identification and segmentation, prior and observation models and error criteria can be very efficiently selected in a statistical framework. Using statistical models in a Bayesian framework enables posing many image processing problems as statistical inference problems.

Among several possible 2-D models for images most of the research has been restricted to Markov random field **(MRF)** models, because of the local statistical dependence of images.

* This work was supported in part by Grant #ASC 9318183 from National Science Foundation. A longer version of this chapter is due to appear in the IEEE Transactions on Image Processing.

0-8186-7644-2 $5.00 © 1996 IEEE

MRF models have been used to characterize prior beliefs about various image features such as textures, edges and region labels. Since MRF models express global statistics in terms of the local neighborhood potentials, all computations are restricted to a local window. This spawned a lot of interest in developing algorithms that utilize local computations to achieve global optimization [9]. But the main drawback is that the optimization schemes associated with MRF energy functions are iterative. Typical MRF algorithms visit all lattice sites in a specific order and perform a local computation at each site; this is repeated until some form of convergence is reached. Even though the individual iterations involve only simple local computations, the iterative nature of these algorithms contributes to the computational burden. Two different approaches have been used to reduce the computational requirement. The first is to use non-optimal, deterministic methods that converge to a local optimal point, but still provide reasonably good results. Geiger and Girosi [8] and Zhang [24] use mean field approximations that lead to deterministic relaxation algorithms. Wu and Doerschuk [23] use a tree approximation that replaces the lattice on which an MRF is defined by an acyclic tree which allows replacing the iterative MRF computations by recursive computations.

The second approach is to use multiresolution techniques. Two important aspects of multiresolution approaches are: (1) divide and conquer and (2) action at a distance [21]. Research efforts on multiresolution models and analysis can be found in [22],[2],[10],[4] and, [16]. We elaborate on the following, because of their relevance to our work.

Jeng, in [12], discusses the effect of subsampling resolution transformation on MRFs and presents two results: first, the Markov property is not preserved for a general subsampling scheme and, second, it is preserved under some specific subsampling schemes depending on the size and shape of the neighborhood. In [15] Lakshmanan and Derin present an excellent discussion on multiresolution GMRF models. It is shown that the GMRFs lose their Markov property under subsampling and expressions for the power spectral density functions at coarser resolution are obtained. It is also shown that for the special case of second order GMRFs with separable autocovariances, the Markov property is retained under subsampling. In addition, a *covariance invariance* approximation is presented to approximate the coarser resolution data by GMRFs. Many interesting properties of this estimator such as maximizing the entropy and minimizing the Kullback-Leibler **(KL)** distance can be found in [15].

We present a multiresolution model based on a KL distance measure. Given that the data at the fine resolution is a GMRF, the goal is to obtain suitable models at coarser resolutions. Data at coarser resolutions are obtained by subsampling the fine resolution data. Under these resolution transformations, coarser resolution data are non-Markov. We present an estimator to compute the parameters corresponding to the "best" GMRF approximation at lower resolutions from the parameters at the fine resolution based on minimizing the KL distance between the conditional densities (conditional relative entropy). We also show that the computations for this estimator turn out to be similar to the psuedo likelihood estimator [1], except that the sample covariances are replaced by covariances calculated with respect to the non-Markov measure that is being approximated. We present results on the existence of different sets of GMRF parameters at fine resolution that result in statistically identical coarser resolution random fields. As an application, we consider the texture segmentation problem and performe segmentation over multiple resolutions using our multiresolution GMRF model. We show that the multiresolution technique performs better than the single resolution approach.

The rest of the chapter is organized as follows. Section 2 introduces the GMRF and the

basics of the resolution transformation. Section 3 presents the Markov approximation for non-Markov fields based on local conditional distribution invariance approximation. Section 4 presents the many-to-one nature of transformation of the GMRF parameters from the fine to coarse resolution. Section 5 presents various aspects of the multiresolution segmentation and Section 6 gives synthetic and real experiments. Section 7 concludes the paper.

2: GMRFs and Resolution Transformation

In this section we introduce basic notations used in the rest of the chapter and also present results on loss of the Markov property under resolution transformation [15].

2.1: The GMRF Model

We use the following notation :
$t = (t_1, t_2), s = (s_1, s_2)$: coordinates of grid points on a 2-D lattice
$\Omega = \{s : 0 \leq s_1 \leq M-1, 0 \leq s_2 \leq N-1\}$: a two dimensional lattice
$\underline{X}$: a random field on Ω, represented as a vector by a row-wise scan ordering
X_s: the random variable at site s
η, ψ, ξ: neighborhood sets

The set of lattice points that are contained in the neighborhood of a site s is denoted by η_s. The elements that are included in the neighborhood of the site marked s for different neighborhood orders can be found in [6].

For the first order case, $\eta = \{(1,0),(0,1),(-1,0),(0,-1)\}$, and $\eta_s = \{s + r : r \in \eta\}$.

If $\underline{X}$ is modeled by a GMRF with a symmetric neighborhood η, then $\underline{X}$ can be written as [13]:

$$X_s = \sum_{r \in \eta} \theta_r X_{s+r} + e_s$$

where $\underline{e}$ is a zero mean, Gaussian noise, with autocorrelation given by :

$$E[e_s e_{s+r}] = \begin{cases} \sigma^2 & \text{if } r = (0,0) \\ -\theta_r \sigma^2 & \text{if } r \in \eta \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Hence the GMRF can be completely characterized by the set of parameters $\{\underline{\theta}, \sigma^2\}$. The parameter set $\underline{\theta}$ should satisfy the following conditions :

$$\begin{aligned} &1. \quad \theta_r = \theta_{-r} \quad \forall r \in \eta \\ &2. \quad 1 - \underline{\theta}^T \underline{\phi}_s > 0 \quad \forall s \in \Omega \end{aligned} \tag{2}$$

where $\underline{\phi}_s$ is a vector whose length is equal to the number of elements in the neighbor set η. The individual elements of $\underline{\phi}_s$ are given by:

$$\cos\left(\left(\frac{2\pi s_1}{M} \frac{2\pi s_2}{N} \right) \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} \right) \quad r \in \eta.$$

The first condition is necessary to ensure stationarity and the second to ensure that the covariance matrix of $\underline{X}$ is positive definite.

$\underline{X}$ exhibits the Markov property [13],

$$\begin{aligned} p(x_s|x_t, \forall t \neq s, t \in \Omega) &= p(x_s|x_{s+r}, r \in \eta) \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{[x_s - \sum_{r\in\eta} \theta_r x_{s+r}]^2}{2\sigma^2}\}. \end{aligned} \tag{3}$$

The power spectrum $S_x(\omega)$ of $\underline{X}$ can be shown to be [13]:

$$S_x(\omega) = \frac{\sigma^2}{1 - \sum_{r\in\eta} \theta_r \cos[\frac{2\pi}{M} r_1\omega_1 + \frac{2\pi}{N} r_2\omega_2]} \tag{4}$$

where $\omega = \{\omega_1, \omega_2\}$, and $0 \leq \omega_1 \leq M-1, 0 \leq \omega_2 \leq N-1$.

2.2: GMRFs and Resolution Transformation

Let $\Omega^{(0)} = \Omega = \{(s_1, s_2) : 0 \leq s_1 \leq M-1, 0 \leq s_2 \leq N-1\}$ be a rectangular lattice and M and N are assumed to be powers of 2. The superscript stands for the level in the image pyramid, $\Omega^{(0)}$ being the lattice at the fine resolution, $\Omega^{(k)}$ represents the lattice which is obtained by subsampling $\Omega^{(0)}$, k times. Let $\underline{X}^{(k)}$ represent a random field, obtained by ordering the random variables on $\Omega^{(k)}$. The parameters of a GMRF defined on a lattice $\Omega^{(k)}$ are denoted by $\{\underline{\theta}^{(k)}, [\sigma^2]^{(k)}\}$ and the associated neighborhood is denoted by $\eta^{(k)}$. The covariance matrix and the power spectrum associated with $\underline{X}^{(k)}$ are denoted by $\Sigma^{(k)}$ and $S_x^{(k)}(\omega)$ respectively. The probability distributions defined on a lattice $\Omega^{(k)}$ are indexed by $p^{(k)}(.)$.

Let $\underline{X}^{(0)}$ be a GMRF defined on $\Omega^{(0)}$ with parameters $\{\underline{\theta}^{(0)}, [\sigma^2]^{(0)}\}$ and a neighborhood $\eta^{(0)}$. The power spectrum of $\underline{X}^{(0)}$ can be written as in Eq. (4) :

$$S_x^{(0)}(\omega) = \frac{[\sigma^2]^{(0)}}{1 - \sum_{r\in\eta^{(0)}} \theta_r^{(0)} \cos[\frac{2\pi}{M} r_1\omega_1 + \frac{2\pi}{N} r_2\omega_2]} \tag{5}$$

where $\omega = \{(\omega_1, \omega_2) : 0 \leq \omega_1 \leq M-1, 0 \leq \omega_2 \leq N-1\}$. The subsampling resolution transformation is defined as:

$$X_s^{(k)} = X_{2s}^{(k-1)}$$

defined for all $s \in \Omega^{(k)}$.

The power spectrum of $\underline{X}^{(k)}$ can be shown to be [15]:

$$S_x^{(k)}(\omega) = \frac{1}{2^{2k}} \sum_{r\in C_k} S_x^{(0)}(\omega + r') \tag{6}$$

where $r' = (\frac{M}{2^k} r_1, \frac{N}{2^k} r_2)$ and $C_k = \{r : 0 \leq r_1 \leq 2^k - 1, 0 \leq r_2 \leq 2^k - 1\}$.

It can be observed that $S_x^{(k)}(\omega)$ cannot be written in the form of Eq. (4) with a finite neighborhood. Therefore, the subsampled fields $\underline{X}^{(k)}$ are non-Markov, except for the special case of second order separable correlation GMRFs [15].

3: Local Conditional Distribution Invariance Approximation

As mentioned in the last section, GMRFs become non-Markov when subsampled. However, if the coarser resolution data are modeled by the exact non-Markov Gaussian measures, conventional optimization techniques based on Markov properties cannot be applied. In this section we show that it is possible to obtain good Markov approximations for coarser resolution fields.

In this section we present a technique to estimate the best GMRF parameters of a non-Markov random field, based on a KL distance measure between local conditional distributions (conditional relative entropy) [5]. In MRF applications all optimizations are performed based on the local conditional distribution, so, we believe an estimator based on it should be well suited for image analysis applications. We also exemplify the connection between this estimator and the pseudo likelihood estimator [1].

The Markov approximation presented in this section is based on linear estimation. Before presenting the details, we will provide a known result regarding the linear estimation of a GMRF. Let $\underline{Z}$ be a GMRF defined by $(\underline{\theta}, \sigma^2)$ with a neighborhood ψ. Then the best estimate of Z_s based on the elements of ψ is given by [3]:

$$\hat{z_s} = \sum_{r \in \psi} \theta_r z_{s+r}$$

and the mean square error

$$E(Z_s - \hat{Z_s})^2 = \sigma^2.$$

The conditional density $p(z_s|z_r, r \in \psi)$ is Gaussian with conditional mean $\sum_{r\in\psi} \theta_r z_{s+r}$ and conditional variance σ^2.

Let $\underline{X}$ be a random field with a stationary non-Markov probability measure $\mathrm{p}(\underline{x})$ *and let* $q^*(\underline{x})$ *be a GMRF approximation such that:*

$$q^*(x_s|x_{s+r}, r \in \eta) = \arg\min_q D[p(x_s|x_{s+r}, r \in \eta) \;||\; q(x_s|x_{s+r}, r \in \eta)], \tag{7}$$

where the minimization is performed over the entire family of GMRF pdfs with a chosen neighborhood η. In addition, under certain conditions (given at the end of the section), $q^*(x_s|x_{s+r}, r \in \eta)$ is exactly equal to $p(x_s|x_{s+r}, r \in \eta)$.

Since $q(\underline{x})$ belongs to the family of GMRF densities, $q(x_s|x_{s+r}, r \in \eta)$ will be of the form given in Eq. (3).

$$q(x_s|x_{s+r}, r \in \eta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{[x_s - \sum_{r\in\eta} \theta_r x_{s+r}]^2}{2\sigma^2}\}.$$

Let $(\underline{\theta}^*, [\sigma^2]^*)$ be the parameters corresponding to $q^*(\underline{x})$. To simplify the notation, let $\underline{Y}$ be the vector containing the neighborhood random variables in a proper order. For a first order neighborhood,

$$\underline{Y}^T = \begin{pmatrix} X_{s+(1,0)} & X_{s+(0,1)} & X_{s+(-1,0)} & X_{s+(0,-1)} \end{pmatrix}.$$

Now performing the minimization in terms of the parameters,

$$(\underline{\theta}^*, [\sigma^2]^*) \;=\; \arg\min_{(\underline{\theta}, \sigma^2)} D[p(x_s|\underline{y}) \;||\; q(x_s|\underline{y})]$$

$$
\begin{aligned}
&= \arg\min_{(\underline{\theta},\sigma^2)} E_p\left[\log\frac{p(X_s|\underline{Y})}{q(X_s|\underline{Y})}\right] \\
&= \arg\max_{(\underline{\theta},\sigma^2)} E_p\left[\log q(X_s|\underline{Y})\right] \\
&= \arg\max_{(\underline{\theta},\sigma^2)} E_p\left[-\frac{1}{2}\log\sigma^2 - \frac{1}{2\sigma^2}(X_s - \sum_{r\in\eta}\theta_r Y_r)^2\right] \\
&= \arg\min_{(\underline{\theta},\sigma^2)} \frac{1}{2}\log\sigma^2 + \frac{1}{2\sigma^2}(E_p[X_s - \sum_{r\in\eta}\theta_r Y_r]^2).
\end{aligned}
\tag{8}
$$

It can be seen that the $\underline{\theta}^*$ parameters corresponding to $q^*(\underline{x})$ are obtained by minimizing the second term in the Eq. (8)

$$
\underline{\theta}^* = \arg\min_{\underline{\theta}} E_p[X_s - \sum_{r\in\eta}\theta_r Y_r]^2 \tag{9}
$$

and using the $\underline{\theta}^*$ obtained, we can estimate the $[\sigma^2]^*$ that minimizes Eq. (8),

$$
[\sigma^2]^* = E_p[X_s - \sum_{r\in\eta}\theta_r^* Y_r]^2. \tag{10}
$$

then,

$$
\begin{aligned}
\underline{\theta}^* &= \arg\min_{\underline{\theta}} E_p[X_s - \underline{\theta}^T\underline{Y}]^2 \\
\underline{\theta}^* &= [E_p(\underline{Y}\underline{Y}^T)]^{-1}E_p(X_s\underline{Y})
\end{aligned}
\tag{11}
$$

and,

$$
\begin{aligned}
[\sigma^2]^* &= E_p(X_s^2) - E_p(X_s\underline{Y}^T)[E_p(\underline{Y}\underline{Y}^T)]^{-1}E_p(X_s\underline{Y}) \\
&= E_p(X_s^2) - [\underline{\theta}^*]^T E_p(X_s\underline{Y}).
\end{aligned}
\tag{12}
$$

In addition, the estimated $\underline{\theta}^*$ parameters should satisfy the positivity conditions in Eq. (2).

Now, returning back to multiresolution discussion, let $X^{(0)}$ be a GMRF defined by $(\underline{\theta}^{(0)}, [\sigma^2]^{(0)})$ and $X^{(k)}$ be the field obtained by subsampling $X^{(0)}$, k times. The non-Markov $X^{(k)}$ can be approximated by a GMRF by minimizing Eq. (7). The minimization requires the autocorrelation values $E_{p^{(k)}}[X_s^{(k)}X_{s+r}^{(k)}]$ which can be computed, given the GMRF parameters for $X^{(0)}$ as shown below.

$$
\begin{aligned}
X_s^{(k)} &= X_{2^k s}^{(0)} \\
E_{p^{(k)}}[X_s^{(k)}X_{s+r}^{(k)}] &= E_{p^{(0)}}[X_{2^k s}^{(0)}X_{2^k(s+r)}^{(0)}].
\end{aligned}
$$

For any two lattice sites u and v in $\Omega^{(0)}$ the correlation is given by [13]:

$$
E_{p^{(0)}}[X_u^{(0)}X_v^{(0)}] = \frac{[\sigma^2]^{(0)}}{MN}\sum_{s\in\Omega^{(0)}}\frac{(\lambda_M^{s_1u_1}\lambda_N^{s_2u_2})(\lambda_M^{-s_1v_1}\lambda_N^{-s_2v_2})}{1 - [\underline{\theta}^{(0)}]^T\underline{\phi}_s} \tag{13}
$$

where $\lambda_n^i = \exp(\sqrt{-1}\frac{2\pi i}{n})$.

Under the assumption that the covariance matrix with respect to p - measure is positive definite, the function in Eq. (9) to be minimized is convex and is minimized over a convex set defined by $1 - [\underline{\theta}^{(k)}]^T \underline{\phi}_s > 0$, for $\forall s \in \Omega$. If the solution lies inside the convex set, it can obtained from Eq. (11). Otherwise, a gradient descent procedure can be used.

Remarks:

1. If the $\underline{\theta}^*$ obtained from Eq. (11) satisfies the positivity conditions and if p is Gaussian, then $p(x_s|x_{s+r}, r \in \eta) = q^*(x_s|x_{s+r}, r \in \eta)$. Since $p(\underline{x})$ is Gaussian, $p(x_s|x_{s+r}, r \in \eta)$ is also Gaussian with conditional mean $\sum_{r\in\eta} \theta_r^* x_{s+r}$ (which is the best linear estimate of X_s in terms of $X_{s+r}, r \in \eta$) and conditional variance $[\sigma^2]^*$ (which is the corresponding minimum mean square error of the estimator) [19]. $q^*(\underline{x})$ being a GMRF with parameters $(\underline{\theta}^*, [\sigma^2]^*)$, from the discussion at the beginning of this section, has the conditional distribution $q^*(x_s|x_{s+r}, r \in \eta)$ with the conditional mean $\sum_{r\in\eta} \theta_r^* x_{s+r}$ and conditional variance $[\sigma^2]^*$. However, the joint densities $p(\underline{x})$ and $q(\underline{x})$ on the whole lattice are not the same, $p(\underline{x})$ is a non-Markov density and $q(\underline{x})$ is a Markov density.

2. It is worth observing that Eq. (8) is similar to the pseudo likelihood estimate [3], [1] where the GMRF parameters are obtained by minimizing the products of local conditional densities over the entire lattice. The pseudo likelihood estimator uses the sample covariances obtained from the observed sample field, whereas our local conditional distribution invariance estimator uses the covariances calculated with respect to the p - measure.

4: Parameters Resulting in Identical PDFs at Coarser Resolutions

In the previous section, we presented methods to approximate subsampled random fields by GMRFs assuming that data at the fine resolution is modeled by a GMRF. It is also necessary to analyze if different GMRF parameters at the fine resolution can result in probabilistically identical coarser resolution random fields. Since we are dealing with Gaussian fields, it suffices to check the covariance matrices of the subsampled fields instead of the pdfs. However, the covariance elements are complicated functions of the parameters (see Eq. (13)). Therefore, we look at the power spectrum of the subsampled random fields which are simpler functions of the parameters. We show that there exists different sets of GMRF parameters, which on subsampling result in the same pdf at the lower resolution. Since the parameter $[\sigma^2]^{(0)}$ is a multiplicative factor in the power spectral function, we assume it be equal to one and investigate the existence of different sets of $\underline{\theta}$ parameters that result in the identical coarser resolution random fields.

Case 1: First order GMRF on $\Omega^{(0)}$

The first order GMRF model is defined by the parameters $(\theta_{(1,0)}, \theta_{(0,1)}, 1)$. For a first order GMRF at the fine resolution, the only set of parameters that results in the same power spectrum at $\Omega^{(1)}$ is $(\theta_{(1,0)}, \theta_{(0,1)})$, $(-\theta_{(1,0)}, \theta_{(0,1)})$, $(\theta_{(1,0)}, -\theta_{(0,1)})$, $(-\theta_{(1,0)}, -\theta_{(0,1)})$.

Case 2: Second order GMRF on $\Omega^{(0)}$

The second order GMRF model is defined by the parameters $(\theta_{(1,0)}, \theta_{(0,1)}, \theta_{(1,1)}, \theta_{(-1,1)}, 1)$. For a second order GMRF at the fine resolution, the only set of parameters that result in the same power spectrum at $\Omega^{(1)}$ is,

$(\theta_{(1,0)}, \theta_{(0,1)}, \theta_{(1,1)}, \theta_{(-1,1)})$, $(-\theta_{(1,0)}, \theta_{(0,1)}, -\theta_{(1,1)}, -\theta_{(-1,1)})$,

$(\theta_{(1,0)}, -\theta_{(0,1)}, -\theta_{(1,1)}, -\theta_{(-1,1)})$, $(-\theta_{(1,0)}, -\theta_{(0,1)}, \theta_{(1,1)}, \theta_{(-1,1)})$.

Proof: The proof can be found in [14].

Similar results can be obtained for higher order cases.

5: Texture Segmentation

Computer vision and image analysis algorithms use various visual cues to analyze and interpret an image of a complex scene. These visual cues include, among others, photometric and geometric cues. Photometric cues include shading, texture, etc., from which features such as edges and regions are obtained. Texture is one of the basic characteristics of a visible surface and provides useful information for scene segmentation and understanding. Texture is a very important property for the analysis of remote sensed satellite images, their segmentation into various vegetation classes. Texture classification and segmentation problems have been addressed by several authors with different approaches that can be broadly classified into two, namely, structural [20],[11] and statistical [6],[7],[18],[17] approaches.

Texture segmentation problem is the labeling of pixels in a lattice to one of V texture classes, based on a texture model and the observed intensity field. Each site in the lattice carries a class label (say $L_s = v, v \in \{1, 2, \ldots, V\}$) and this label field is modeled by an MRF. We do not directly observe the label field, but a function of the labels, the intensity field. The intensity field is modeled by a GMRF, whose parameters depend on the value of label field at that site. The goal is to estimate the unobserved label field from the observed intensities by optimizing a suitable error criterion.

We model the label field $\underline{L}$ by an MRF with a neighborhood ψ:

$$p(\underline{L} = \underline{l}) = \frac{1}{Z} \exp\left[\beta \sum_{s \in \Omega} U(l_s)\right] \tag{14}$$

where $U(l_s)$ is the number of neighbors in ψ that have the same label as l_s. This model is also called a pairwise interaction model.

The local conditional probability of the label field is given by:

$$p(l_s | l_{s+r}, r \in \psi) = \frac{\exp\left[\beta U(l_s)\right]}{\sum_{l'_s=(1,2,\ldots V)} \exp\left[\beta U(l'_s)\right]}. \tag{15}$$

The GMRF parameters corresponding to a label v are denoted by $(\underline{\theta}(v), \sigma^2(v))$. The conditional density of the intensity field can be written as follows, from Eq. (3):

$$\begin{aligned} & p(X_s = x_s | L_s = v, X_{s+r}, r \in \eta) \\ &= \frac{1}{\sqrt{2\pi\sigma^2(v)}} \exp\{-\frac{1}{2\sigma^2(v)}[x_s - \sum_{r \in \eta} \theta_r(v) x_{s+r})]^2\}. \end{aligned} \tag{16}$$

We restrict ourselves to the iterated conditional mode method (ICM) . The ICM solution is obtained by performing the following optimization at each lattice site [1]:

$$\max_{L_s} \quad P(X_s | L_s, X_{s+r}, r \in \eta) P(L_s | L_{s+r}, r \in \psi).$$

This is equivalent to,

$$\min_{L_s} \quad \frac{1}{2} \log(\sigma^2(v)) + \frac{1}{2\sigma^2(v)}[x_s - \sum_{r \in \eta} \theta_r(v) x_{s+r}]^2 - \beta U(L_s = v) \tag{17}$$

the minimization is performed by visiting the pixels in raster scan order for all $s \in \Omega$ and stopped when no further changes in the labels occur.

5.1: Multiresolution Segmentation

The segmentation algorithm presented above is a single resolution algorithm. As we have discussed before, data at lower resolutions can be approximated by a GMRF. Thus the same algorithm can be applied at lower resolutions too. Our multiresolution algorithm includes the following steps. First, given the number of classes and the associated parameters at the fine resolution, the GMRF parameters at lower resolutions are obtained by the local conditional distribution invariance approximation. Then, segmentation is performed at the coarsest resolution using Eq. (17) with the corresponding parameters and the results of segmentation are passed on to the immediate higher resolution. This is repeated until the fine resolution is reached. At each resolution a confidence measure is attached to the segmentation result at each pixel and propagated to the finer resolution. We address issues regarding confidence measures in this section. After obtaining the segmentation result by ICM convergence at one resolution, the results have to be propagated to the immediate higher resolution. Since we obtain resolution transformation by subsampling, we have a quad tree type of graph. If $\underline{L}^{(k)}$ is the segmentation result at the kth resolution, the labels in the $(k-1)$th level are initialized as:

$$L_s^{(k-1)} = L_{\lfloor s/2 \rfloor}^{(k)}. \tag{18}$$

In addition, at level k, after the ICM converges, we attach a confidence measure $C_s^{(k)}$ to the segmentation result obtained at site s.

At level k, after the convergence of ICM iterations, let $\dot{\underline{v}}$ and $\ddot{\underline{v}}$ be such that,

$$\begin{aligned} \dot{v}_s &= \arg \max_{v \in \{1,2,\ldots,V\}} P(X_s|L_s = v, X_{s+r}, r \in \eta) P(L_s = v|L_{s+r}, r \in \psi) \\ \ddot{v}_s &= \arg \max_{v \in \{1,2,\ldots,V\}, v \neq \dot{v}_s} P(X_s|L_s = v, X_{s+r}, r \in \eta) P(L_s = v|L_{s+r}, r \in \psi) \end{aligned}$$

and the confidence measure is defined as,

$$C_s^{(k)} = \frac{P(X_s|\dot{v}_s, X_{s+r}) P(\dot{v}_s|L_{s+r})}{P(X_s|\ddot{v}_s, X_{s+r}) P(\ddot{v}_s|L_{s+r})}. \tag{19}$$

These confidence measures at level k are propagated upwards to level $k-1$ in the same manner as in Eq. (18). At level k, ICM is restricted to only those pixels with the confidence measure such that, $\frac{1}{C_s^{(k)}} \geq c^{(k)}$, where $c^{(k)}$ is a confidence threshold at level k. Also from the definition, $0 < \frac{1}{C_s^{(k)}} \leq 1.0$. For the coarsest resolution $c^{(\cdot)} = 0$, i.e., ICM is performed over all sites in the lattice.

6: Experiments

We present experimental results with simulated, Brodatz texture images and real satellite images to show that the multiresolution algorithms perform better than the single resolution both in terms of the classification accuracy and computational requirements. In all the experiments, the confidence threshold $c^{(k)} = \{0.6, 0.25, 0.0\}$, is used for the different levels with smaller values used at coarser resolutions. Multiresolution results presented in this section are obtained by performing the algorithm over three resolutions. In all cases, percentages of correct classification and computational requirements are given in parenthesis.

To compare the computational requirements between the single resolution and multiresolution approaches, we define a unit of computation to be the computation required to perform ICM at a single pixel site.

We generated texture images using the technique given in [3]. Three third order GMRF textures are generated with parameters { ($\theta_{(1,0)}$ = 0.0934154, $\theta_{(0,1)}$ = 0.520252, $\theta_{(1,1)}$ = 0.0303413, $\theta_{(-1,1)}$ = 0.0180476, $\theta_{(2,0)}$ = -0.0216434, $\theta_{(0,2)}$ = -0.148331), σ^2 = 0.9342 }, {$\underline{\theta}$= (0.308257, 0.468389, -0.0755398, -0.0755797, -0.0407557, -0.100678), σ^2 = 1.8472 }, {$\underline{\theta}$= (0.406875, 0.423393, -0.178478, -0.188702, -0.0649544, -0.121439), σ^2 = 1.264811 }. Figure 1(a) shows the composite image with these three textures. Figure 1(b) shows the single resolution segmentation result (classification accuracy = 89.84%, computational requirement = 2686976) and Figure 1(c) shows the result for multiresolution segmentation (96.75%, 431031).

We have tested our algorithm on textures from the Brodatz texture album. Figure 2(a) contains grass, calf leather, wool, and wood textures. The original GMRF parameters are estimated by maximum likelihood estimation. Figure 2(b) shows the single resolution segmentation (86.04%, 1114112) and Figure 2(c) shows the multiresolution segmentation (92.75%, 679444). We have another interesting plot of $\frac{1}{C_s^{(k)}}$ for the level $k = 1$ in Figure 2(d). The brighter points in this image correspond to points of low confidence measure. As expected, all the boundary regions between different textures have low confidence measures. In texture segmentation, classification near the texture boundaries is usually more ambiguous.

Figure 3(a) shows a section of a single channel of a multispectral sensor (MSS) image over Africa. We chose three classes corresponding to river, forest, and deforestation. The GMRF parameters obtained from small sections of a different part of the image are used to classify the image shown. Unfortunately exact class maps are not available. Figure 3(b) shows the single resolution result (unknown, 2160000) and Figure 3(c) shows the multiresolution result (unknown, 426105). Clearly, we can see that the multiresolution algorithm has performed better, with lesser computation, than the single resolution algorithm.

Finally, we present results of multiresolution segmentation on a thematic mapper (TM) image consisting of four classes corresponding to river, forest, deforestation and regrowth. Figure 4(a) shows a section of thematic mapper (TM) data and Figure 4(b) shows the 4-class multiresolution segmentation result.

7: Summary

Multiresolution models and algorithms play an important role in image analysis. These algorithms not only help to reduce the computational time, but also help to analyze the given information at different spatial scales. We have presented a technique based on minimizing the KL distance, to estimate the parameters of GMRFs at coarser resolutions and have used it for texture segmentation. GMRFs are widely used in many image processing applications including restoration, segmentation, compression, etc., and the proposed models can be used for these applications. Also, this can be extended to perform unsupervised texture segmentation. However, as mentioned in Section 4, GMRF parameters at a lower resolution can correspond to more than one set of parameters at fine resolution. Hence the problem of retrieving the GMRF parameters at fine resolution given the parameters at coarse resolution has to be addressed for unsupervised segmentation.

References

[1] J. Besag, "On the Statistical Analysis of Dirty Pictures," *Journal of the Royal Statistical Society*, Vol. 48, pp. 259–302, 1986.

[2] C. Bouman and B. Liu, "Multiple Resolution Segmentation of Textured Images," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 13, pp. 99–113, Feb. 1991.

[3] R. Chellappa, "Two-dimensional Discrete Gaussian Markov Random Field Models for Image Processing," in *Progress in Pattern Recognition* (L. N. Kanal and A. Rosenfeld, eds.), pp. 79–112, Elsavier, 1985.

[4] F. S. Cohen and D. B. Cooper, "Simple Parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal Markovian Random Fields," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 9, pp. 195–219, March 1987.

[5] T. Cover and J. Thomas, *Elements of Information Theory*, Wiley, 1991.

[6] G. R. Cross and A. K. Jain, "Markov Random Field Texture Models," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 5, pp. 25–39, Jan 1983.

[7] H. Derin and H. Elliot, "Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Field," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 9, pp. 39–55, Jan. 1987.

[8] D. Geiger and F. Girosi, "Parallel and Deterministic Algorithms for MRFs: Surface Reconstruction," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 13, pp. 401–413, May 1991.

[9] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 6, pp. 721–741, Nov 1984.

[10] B. Gidas, "A Renormalization Group Approach to Image Processing," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 11, No. 2, pp. 164–180, 1989.

[11] R. Haralick, "Statistical and Structural Approaches to Texture," *Proc. IEEE*, Vol. 67, pp. 610–621, May 1979.

[12] F. C. Jeng, "Subsampling of Markov Random Fields," *Jour. of Visual Communication and Image Representation*, Vol. 3, pp. 225–229, Sep. 1992.

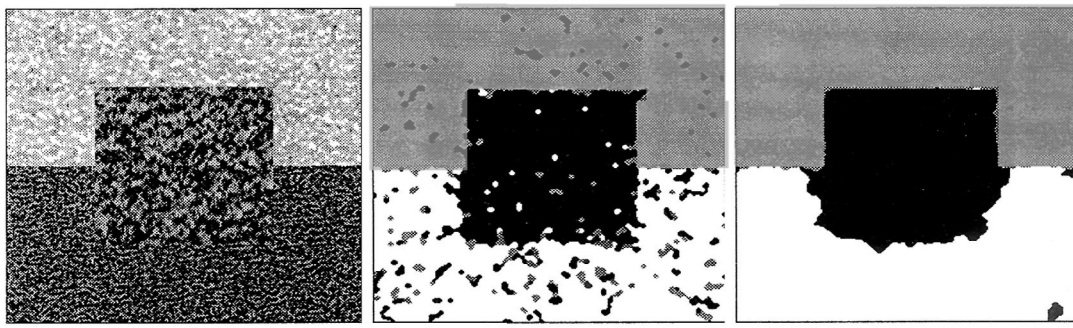
[13] R. L. Kashyap, "Analysis and Synthesis of Image Patterns by Spatial Interaction Models," in *Progress in Pattern Recognition* (L. N. Kanal and A. Rosenfeld, eds.), North-Holland, Amsterdam, 1981.

[14] S. Krishnamachari, *Hierarchical Markov Random Field Models for Image Analysis*, Ph.D. dissertation, University of Maryland, College Park, 1995.

[15] S. Lakshmanan and H. Derin, "Gaussian Markov Random Fields at Multiple Resolutions," in *Markov Random Fields: Theory and Applications* (R. Chellappa, ed.), pp. 131–157, Academic Press, 1993.

[16] M. R. Luettgen, W. C. Karl, A. S. Willsky, and R. R. Tenney, "Multiscale Representations of Markov Random Fields," *IEEE Trans. on Signal Processing*, Vol. 41, pp. 3377–3397, Dec. 1993.

[17] B. S. Manjunath and R. Chellappa, "A Note on Unsupervised Texture Segmentation," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 13, pp. 478–483, May 1991.



(a) (b) (c)

Figure 1: (a) Synthetic texture image, (b) Single resolution segmentation result, (c) Multiresolution segmentation result.

[18] B. S. Manjunath, T. Simchony, and R. Chellappa, "Stochastic and Deterministic Networks for Texture Segmentation," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. 38, pp. 1039–1049, June 1990.

[19] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill Book Company, 1965.

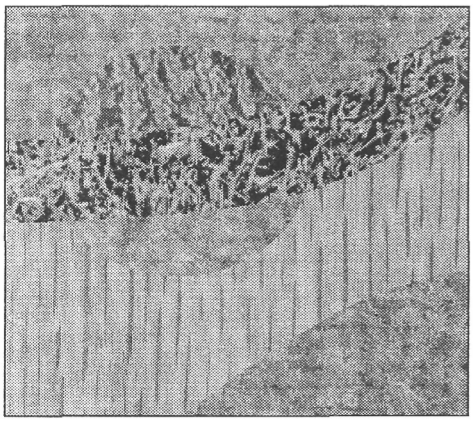
[20] A. Rosenfeld, "Visual Texture Analysis," Tech. Rep. 70-116, University of Maryland, June 1970.

[21] A. Rosenfeld, "Some Useful Properties of Pyramids," in *Multiresolution Image Processing and Analysis* (A. Rosenfeld, ed.), Springer-Verlag, 1984.

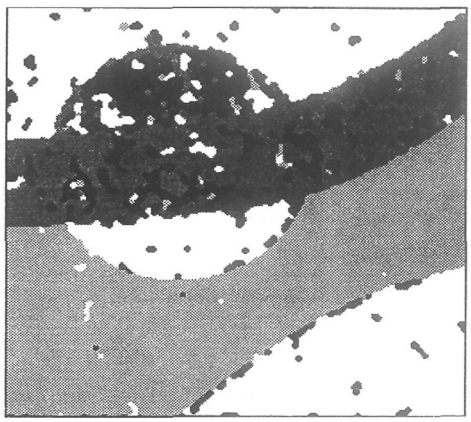
[22] D. Terzopoulos, "Image Analysis using Multigrid Relaxation Methods," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 8, pp. 129–139, March 1986.

[23] C. H. Wu and P. C. Doerschuk, "Tree Approximations to Markov Random Fields," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 17, pp. 391–343, Apr. 1995.

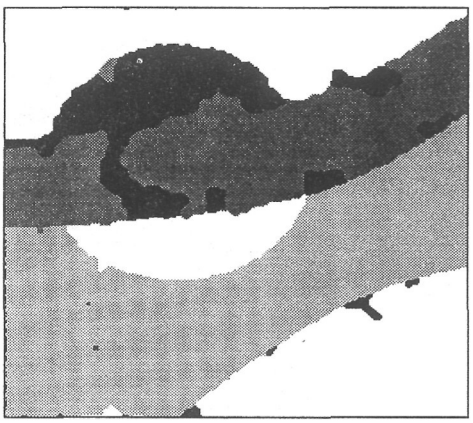
[24] J. Zhang, "The Mean Field Theory in EM Procedures for Markov Random Fields," *IEEE Trans. on Signal Processing*, Vol. 40, pp. 2570–2583, Oct. 1992.



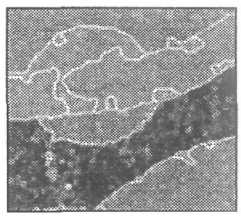
(a)



(b)



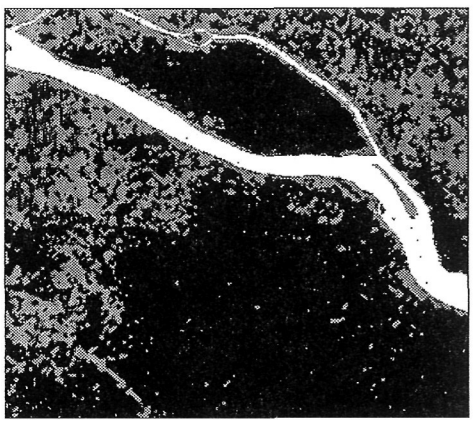
(c)



(d)

Figure 2: (a) Brodatz texture image, (b) Single resolution segmentation result, (c) Multiresolution segmentation result, (d) Confidence measures.

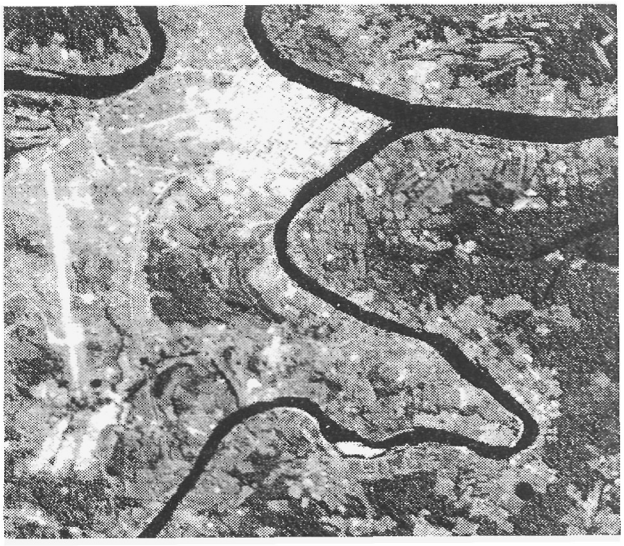
(a)



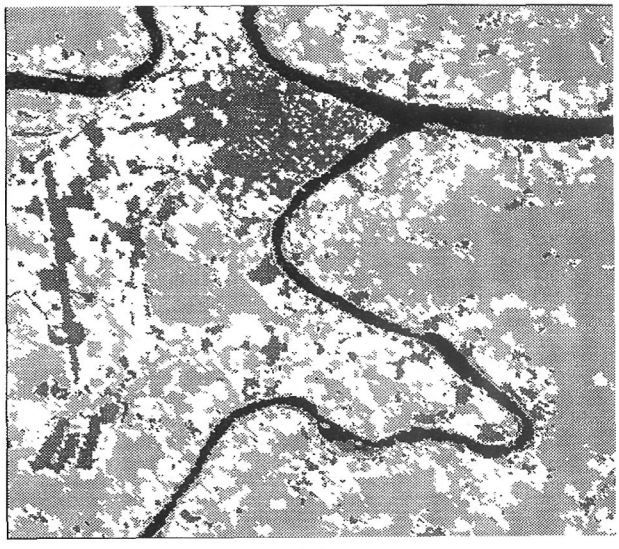
(b)

(c)

Figure 3: (a) Remotely sensed MSS image, (b) Single resolution segmentation result, (c) Multiresolution segmentation result.



(a)



(b)

Figure 4: (a) Remotely sensed TM image, (b) Multiresolution segmentation result.

Feature Selection for Texture Segmentation

Michael Brady and Zhi-Yan Xie

1: Abstract

Wavelet transforms are attracting increasing interest in computer vision because they provide a mathematical tool for multiscale image analysis. We develop a wavelet-based approach to segmenting textured images and demonstrate its application to a range of natural images. First, we investigate the wavelet transform representation of an image at each scale and across scales. We show first that the subsampled wavelet multiresolution representation is translationally variant. More importantly, we show that a wavelet transform of a signal generally confounds the phase component of the analysing wavelet associated with that scale and orientation. The importance of this observation is that commonly used features in texture analysis, such as squaring, or half-, full-wave rectification of a wavelet transform, also depend on this phase component. This not only causes unnecessary spatial variation of features at each scale but also makes it more difficult to match features across scales.

As the main contribution of the paper, we propose a complete 2D decoupled local energy and phase representation of a wavelet transform. As a texture feature, local energy is not only immune to spatial variations caused by the phase component of the analysing wavelet, but facilitates the analysis of similarity of across scales. The success of the approach is demonstrated by experimental results for aerial Infrared Line Scan (IRLS) aerial, satellite, and Brodatz images.

2: Introduction

Texture is a rich source of visual information about the nature and three-dimensional shape of physical surfaces. Following the pioneering work of Azriel Rosenfeld [24], computer texture analysis is ultimately concerned with automated methods to derive such information using artificial systems. There are three major issues in texture analysis:

1. Texture discrimination - to partition a textured image into regions, each corresponding to a perceptually homogeneous texture;
2. Texture classification - to determine to which of a finite number of physically defined classes, such as wood or water, a homogeneous texture region belongs;
3. Shape from texture: to derive 3D surface geometry from texture information.

This paper is confined to texture discrimination- the first stage which has subsequently to be relied on by texture classification and by shape from texture.

Approaches to texture analysis are commonly divided into structural, statistical, model-based and transform methods.

Structural approaches [24, 1] represent textures by well-defined primitives (microtexture) and a hierarchy of spatial arrangements (macrotexture) of those primitives. Although there

0-8186-7644-2 $5.00 © 1996 IEEE

is much benefit to be had from abstract descriptions of textures, they can still be ill-defined for natural textures because the variability of both microtexture and macrotexture means there is no clear distinction between them.

In contrast to structural methods, statistical approaches do not attempt to understand "explicitly" the hierarchical structure of a texture, such as the primitives used in structural approaches. Instead they represent a texture indirectly by the nondeterministic properties which govern the distributions and relationships between the grey levels of an image. Methods based on second-order statistics [3, 4, 5, 13] (i.e. the statistics given by pairs of pixels) have been shown to achieve higher discrimination rates than the power spectrum and structural methods [25, 23]. However, such grey level statistics are of very limited descriptive power and have not given satisfactory results in practice.

Model based image texture analysis [9, 11, 20], using stochastic and fractal models, attempt to interpret an image texture by use of a generative image model or stochastic model. The parameters of the model are estimated and used for texture analysis. In practice, the computational complexity arising in the estimation of stochastic model parameters and the difficulties of handling non-stationary textures are the primary problems. Although such methods have had some success in supervised texture segmentation and classification, in general they work poorly for nonsupervised texture segmentation. The fractal model has been shown to be useful for modelling naturally occurring textures. However, although it has been shown to be useful for texture segmentation [22, 21, 27], it lacks orientation selectivity and is incomplete for describing local image structures.

Transform methods of texture analysis, such as the Fourier [2], Gabor [12, 10, 6, 15, 17], and wavelet transforms [19, 8, 16], represent an image in a coordinate system that has an interpretation that is closely related to the characteristics of a texture, such as frequency or size. Methods based on the Fourier transform perform poorly in practice due to its lack of spatial localisation. Although the Gabor transform can overcome the spatial localisation problem to some extent, it is limited in practice because there is usually not a single resolution at which one can localise spatial structures that form natural textured images. For this reason, the wavelet transform offers the best hope.

A continuous wavelet transform (CWT) is defined by

$$W_\psi f(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} f(x)\overline{\psi}(\frac{x-b}{a})\,dx \tag{1}$$

where $a \in \mathbf{R}^+, b \in \mathbf{R}$ are scale and translation parameters and the wavelet $\psi(x)$ can be any function that satisfies the invertibility condition, i.e. it has to be bandpass. By varying the scale a, $W_\psi f(a,b)$ can capture not only large (low frequency) but also localised (high frequency) components of spatial structures.

Compared with the Gabor transform, the wavelet transforms enjoy several advantages: i) varying the spatial resolution allows it to represent spatial structures at the most suitable scale; ii) there is a wide range of choices for the wavelet function $\psi(x)$; and iii) the flexibility of being able to choose a single (or a set of) wavelet function best suited to the image texture in a specific application. All of these make the wavelet transform particularly attractive for texture segmentation. We begin in Sections 3, 4 by pointing out two theoretical and practical difficulties of using wavelets. Then in Section 5 we define a local energy model that solves these problems. We show typical results on real textured images in Section 6.

3: Translation Invariant Wavelet Multiresolution representation

If the scale and space parameters of CWT are sampled at $\{a = 2^{-j};\ b = 2^{-j}n;\ j, n \in \mathbf{Z}\}$, and if the wavelet $\psi(x)$ is bi-orthogonal, then the wavelet transform is given by

$$\begin{aligned} W_\psi f(j,n) &= 2^{\frac{j}{2}} \int_{-\infty}^{\infty} f(x)\overline{\psi}(2^j x - n)\, dx \\ &\triangleq < f(x), \psi_{j,n}(x) > \end{aligned} \tag{2}$$

where $\psi_{j,n}(x) = 2^{\frac{j}{2}}\psi(2^j x - n)$. $W_\psi f(j,n)$ defines a class of discrete wavelet transforms, the so-called wavelet multiresolution representation (WMR), which can be computed by recursive filtering.

3.1: Translation Invariant WMR

The WMR has been shown useful in image compression because of its completeness and compactness. However, $W_\psi f(j,n)$ is translationally variant:

$$\begin{aligned} W_\psi f(j,n) &= 2^{\frac{j}{2}} < f(x - x_0), \psi_{j,n}(x) > \\ &= 2^{\frac{j}{2}} < f(x), \psi_{j,n-2^j x_0}(x) > \\ &\neq 2^{\frac{j}{2}} < f(x), \psi_{j,n-m}(x) > \end{aligned}$$

for an arbitrary translation x_0, $x_0 = 2^{-j}m$; $m \in \mathbf{Z}$ won't in general be satisfied. In other words, if two identical signals were to appear in different positions, their wavelet transform representations can be very different which is unacceptable for most signal and image processing applications, in particular it badly affects texture segmentation.
One way to overcome this problem is to avoid the scale-dependent subsampling $b = 2^{-j}n$ by setting $b = n$, that is by keeping the same number of samples as the original signal. More precisely, for a signal $f(n)$ the wavelet transform is redefined to be

$$W_\psi f(j,n) = 2^j \sum_k f(k)\overline{\psi}(2^j(k-n))$$

In this way, $W_\psi f(j,n)$ becomes translationally *invariant*.
The 1D wavelet multiresolution representation has been extended to 2D [19] which is defined by

$$\begin{aligned} A^j(m,n) &= < f(x,y), 2^{2j}\phi(2^j x - m)\phi(2^j y - n) > \\ D_x^j(m,n) &= < f(x,y), 2^j\phi(2^j x - m)2^j\psi(2^j y - n) > \\ D_y^j(m,n) &= < f(x,y), 2^j\psi(2^j x - m)2^j\phi(2^j y - n) > \\ D_d^j(m,n) &= < f(x,y), 2^j\psi(2^j x - m)2^j\psi(2^j y - n) > \end{aligned} \tag{3}$$

The mother wavelets associated with $D_x^j(m,n), D_y^j(m,n), D_d^j(m,n)$ are

$$\Psi_x(x,y) = \phi(x)\psi(y) \tag{4}$$
$$\Psi_y(x,y) = \psi(x)\phi(y) \tag{5}$$
$$\Psi_d(x,y) = \psi(x)\psi(y) \tag{6}$$

where ϕ and ψ are the 1D scaling (low-pass) and wavelet function (high-pass), respectively. D_x, D_y, D_d are called the horizontal, vertical and diagonal detail images (channels), respectively, which give strong responses to spatial structures in the horizontal, vertical and diagonal directions[1].
Similarly, it can be shown that A, D_x, D_y, D_d are translationally *variant*. To overcome this problem, we change scale-dependent sampling to uniform sampling as in 1D, but in both the x and y directions. More precisely, an *oversampled* 2D wavelet multiresolution representation is defined by

$$\begin{aligned} A^j(m,n) &= < f(x,y), 2^j\phi(2^j(x-m))2^j\phi(2^j(y-n)) > \\ D_x^j(m,n) &= < f(x,y), 2^{2j}\Psi_x(2^j(x-m), 2^j(y-n)) > \\ D_y^j(m,n) &= < f(x,y), 2^{2j}\Psi_y(2^j(x-m), 2^j(y-n)) > \\ D_d^j(m,n) &= < f(x,y), 2^{2j}\Psi_d(2^j(x-m), 2^j(y-n)) > \end{aligned} \tag{7}$$

in which case A, D_x, D_y, and D_d become translationally *invariant*.
Although the representation given in Eqn. 7 sacrifices the compactness compared with the representation given by Eqn. 3, they provide translational invariance which is essential for texture segmentation, and many other image processing tasks.

4: Phase Dependence of The Wavelet Transform

Considered as a linear convolution with a set of bandpass filters, the wavelet transform enables the image "gradient" to be computed at multiple spatial scales. However the wavelet transform coefficients themselves are not suitable as texture features because they are always zero mean at each scale since $\int_{-\infty}^{\infty} \psi(x)\,dx = 0$. Therefore some non-linear operation is necessary [18].
The most commonly used features are full- and half-wave rectification and the square power of the wavelet detail signals. We have shown [28], however, that in general those features are coupled with the local phase component that depends not only on the analysed signal but also on the analysing wavelet at that scale. This dependency causes two problems that greatly affect practical texture segmentation: "spurious" spatial variations of features at each scale; and the difficulty of matching features across scales.

To illustrate, consider the oversampled 2D wavelet transform of a synthetic images shown in Figure 1. The surfaces of $D_x^j(m,n), D_y^j(m,n), D_d^j(m,n)$ oscillate in space depending on the shape of the wavelets. Patches of moduli $|D_d^j(m,n)|$ at different scales are shown in Figure 2 (top row), and they are clearly affected by the oscillation of the wavelet. Such performance is unacceptable for texture analysis because one wants a uniform feature response in those regions of the image which have uniform texture, while a wavelet is typically an oscillating, wave-like function. Hence, some other nonlinear operation must be found to derive features which can be invariant to *the phase component at each scale* and can also be *matched from one scale to another.*
In 1D, we have previously [26, 28] developed a decoupled local energy and phase representation of a real-valued wavelet transform using the Hilbert transform [28]. In the following section, we show how to extend this representation to 2D.

[1] The orientation of the spatial structure is defined as perpendicular to the direction of maximum gradient.

5: Decoupled 2D Local Energy and Phase Representation

The principal theoretical difficulty in extending the local energy and phase representation of the 1D wavelet transform to 2D is that there does not exist a universal 2D Hilbert transform. Nevertheless, we propose a definition for the local energy and local phase of a 2D wavelet transform which not only provides a complete representation of a 2D wavelet transform in scale-space, but also facilitates the local energy to be independent of the phase components of the analysing wavelets. Moreover, the relationship between a 2D wavelet transform and its local energy is established both in *scale-space* and in *frequency* space.

5.1: Horizontal and Vertical Channels

Recall the mother wavelets associated with the horizontal and vertical channels given in Eqns: 4,5. $D_x^j(m,n)$ can be considered as a 1D wavelet transform with respect to $\psi(y)$ for each column (y axis) after first smoothing each row (x axis) with $\phi(x)$. Similarly, $D_y^j(m,n)$ can be considered as a 1D wavelet transform for each row after first smoothing each column. In these case, the local energy and phase can be defined as 1D [28]:

Definition 5.1 *For a real valued ψ and $f(x,y) \in \mathbf{L^2(Z^2)}$ the local energy ρ_x, ρ_y and the local phase φ_x, φ_y of $D_x^j(m,n)$ and $D_y^j(m,n)$ are given by*

$$\begin{aligned}
\rho_x^j(m,n) &= \sqrt{[D_x^j(m,n)]^2 + [H_y\{D_x^j(m,n)\}]^2} \\
\rho_y^j(m,n) &= \sqrt{[D_y^j(m,n)]^2 + [H_x\{D_y^j(m,n)\}]^2} \\
\varphi_x^j(m,n) &= Atan2\frac{H_y\{D_x^j(m,n)\}}{D_x^j(m,n)} \\
\varphi_y^j(m,n) &= Atan2\frac{H_x\{D_y^j(m,n)\}}{D_y^j(m,n)}
\end{aligned}$$

where $H_x\{.\}$ ($H_y\{.\}$) denotes the Hilbert transform of the 1D function $D(m,n)$ when n (m) is fixed.

5.2: Diagonal Channel

The mother wavelet associated with the diagonal channel $D_d^j(m,n)$ is $\Psi_d(x,y) = \psi(x)\psi(y)$. In this case, if ψ_H is the Hilbert transform of ψ, we construct four complex functions as follows

$$G_1(x,y) = [\psi(x) + i\psi_H(x)][\psi(y) + i\psi_H(y)] \quad (8)$$
$$G_2(x,y) = [\psi(x) - i\psi_H(x)][\psi(y) - i\psi_H(y)] \quad (9)$$
$$G_3(x,y) = [\psi(x) + i\psi_H(x)][\psi(y) - i\psi_H(y)] \quad (10)$$
$$G_4(x,y) = [\psi(x) - i\psi_H(x)][\psi(y) + i\psi_H(y)] \quad (11)$$

Noticing the conjugacy relationships

$$G_1^j(m,n) = \overline{G}_2^j(m,n) \qquad G_3^j(m,n) = \overline{G}_4^j(m,n)$$

only one pair $\{G_k^j(m,n); k = 1,3\}$ or $\{G_k^j(m,n); k = 2,4\}$ needs to be considered. In the following, we use the first pair. Substituting $\Psi_d(x,y)$ with $\{G_k(x,y); k = 1,3\}$ in Eqn. 7,

we generate two complex images by:

$$DG_k^j(m,n) \stackrel{\text{def}}{=} < f(x,y), 2^{2j}G_k(2^j(x-m), 2^j(y-n)) > ;\ k=1,3 \quad (12)$$

$$= (f(x,y) * 2^{2j}\overline{G}_k(-2^j x, -2^j y)) \quad (13)$$

The properties of functions $\{DG_k^j(m,n); k=1,3\}$ are essential for deriving the decoupled local energy and local phase representation of the diagonal channel. We present them in the following lemma.

Lemma 5.1 *For each scale j The functions $DG_1^j(m,n)$ and $DG_3^j(m,n)$ can be represented by $D_d^j(m,n)$ as*

$$DG_1^j(m,n) = D_d^j(m,n) - H_y\{H_x\{D_d^j(m,n)\}\} + i(H_x\{D_d^j(m,n)\} + H_y\{D_d^j(m,n)\}) \quad (14)$$

$$DG_3^j(m,n) = D_d^j(m,n) + H_y\{H_x\{D_d^j(m,n)\}\} + i(H_x\{D_d^j(m,n)\} - H_y\{D_d^j(m,n)\}) \quad (15)$$

and they give a strong response to spatial structures at, or close to $\frac{1\pi}{4}$ and $\frac{3\pi}{4}$, respectively.

Proofs: see Appendix.

Now we are in the position to define the local energy and local phase of the diagonal channel.

Definition 5.2 *The functions $DG_1^j(m,n)$, $DG_3^j(m,n)$ are complex functions and can be written as*

$$DG_1^j(m,n) = \rho_{x+y}^j(m,n)e^{i\varphi_{x+y}^j(m,n)}$$

$$DG_3^j(m,n) = \rho_{x-y}^j(m,n)e^{i\varphi_{x-y}^j(m,n)}$$

The $[\rho_{x+y}^j(m,n)]^2$ and $\varphi_{x+y}^j(m,n)$ are called the local energy and the local phase of $D_d^j(m,n)$, respectively at or close to $\frac{\pi}{4}$. The $[\rho_{x-y}^j(m,n)]^2$ and $\varphi_{x-y}^j(m,n)$ are called the local energy and the local phase of $D_d^j(m,n)$, respectively, at or close to $\frac{3\pi}{4}$.

Now we have defined four local energy channels for each scale j, denoted by $\rho_x^j(m,n)$, $\rho_y^j(m,n)$, $\rho_{x+y}^j(m,n)$, $\rho_{x-y}^j(m,n)$, which are oriented in the horizontal, vertical and $\frac{\pi}{4}$, $\frac{3\pi}{4}$ directions, respectively. Comparing the local energy and the wavelet detail images shown in Figure 1, the phase dependency embedded in the wavelet detail images has been removed in the local energy representations. Moreover, the local energy images at different scales become comparable in terms of shape similarity as indicated by the correlation of the images adjacent scales shown in Figure 2 (bottom row).

5.3: Properties of The Local Energy and Local Phase

The following theorem shows that the local energy and local phase defined above provides a complete representation of the 2D wavelet transform. The local energy and the wavelet transform are equivalent in the frequency domain (conserve energy), but they are very different in scale-space.

Theorem 5.1 *For a real valued $\psi(x)$ and $f(x,y) \in \mathbf{L^2(Z^2)}$,*

1. *The wavelet transform $D_x^j(m,n)$, $D_y^j(m,n)$ and $D_d^j(m,n)$ can be represented completely by the local energies and local phases and are given by*

$$D_x^j(m,n) = \rho_x^j(m,n)\cos\varphi_x^j(m,n) \quad (16)$$

$$D_y^j(m,n) = \rho_y^j(m,n)\cos\varphi_y^j(m,n) \quad (17)$$

$$D_d^j(m,n) = \frac{1}{2}(\rho_{x+y}^j(m,n)\cos\varphi_{x+y}^j(m,n) + \rho_{x-y}^j(m,n)\cos\varphi_{x-y}^j(m,n)) \quad (18)$$

2. *For each scale $j < 0$.*

$$\sum_m\sum_n[D_x^j(m,n)]^2 = \frac{1}{2}\sum_m\sum_n[\rho_x^j(m,n)]^2 \quad (19)$$

$$\sum_m\sum_n[D_y^j(m,n)]^2 = \frac{1}{2}\sum_m\sum_n[\rho_y^j(m,n)]^2 \quad (20)$$

$$\sum_m\sum_n[D_d^j(m,n)]^2 = \frac{1}{8}\sum_m\sum_n([\rho_{x+y}^j(m,n)]^2 + [\rho_{x-y}^j(m,n)]^2) \quad (21)$$

Proofs: see Appendix.
The wavelet detail images and their associated local energy images are very different in scale-space: the former confounds the phase component, the latter does not. Further, from Eqns. 16, 17, 18, it is clear that full-, half-wave rectification or squaring of the wavelet transform also confound the phase component. The difference between the modulus and the local energy of the wavelet transform are shown in Figure 2.

6: Application to Texture Segmentation

To overcome the phase dependency and spatial localisation problems, we suggest that the local energies be used as local features. Using these features, we develop a computation scheme that is used for texture segmentation. The scheme is depicted in Figure 3.

At the **first level**, the 2D **oversampled wavelet transform** is applied to an image. This transform decomposes an image into a stack of images, each of which is given by an oversampled wavelet detail image denoted by $D(\theta,j,x,y)$ at sampled orientation $\theta = \{\theta_1,\cdots,\theta_n\}$ and sampled scale $a = \{2^{-j};\ j = -1,-2,\cdots,-J\}$. For a 2D separable wavelet transform, an image is decomposed into a pile of images $\{D_x^j(x,y), D_y^j(x,y), D_d^j(x,y); j = -1,-2,\cdots,-J\}$.
The **second level** is a nonlinear operation to remove the phase dependency from each image $D(\theta,j,x,y)$, to obtain a pile of local energy images $\rho(\theta,j,x,y)$ given by

$$\rho(\theta,j,x,y) = \begin{cases} \rho_x^j(x,y) & \theta = 0 \\ \rho_y^j(x,y) & \theta = \frac{\pi}{2} \\ \rho_{x+y}^j(x,y) & \theta = \frac{\pi}{4} \\ \rho_{x-y}^j(x,y) & \theta = \frac{3\pi}{4} \end{cases} \quad (22)$$

This level operates only within a single scale, hence it is also called **intra-scale nonlinear fusion**. The local energy images of a Infrared line scan (IRLS) aerial images are shown in Figure 4.
The **third level** derives two texture features in wavelet scale-space, i.e. a multi-scale

orientational measure $\alpha(j,x,y)$ and an energy measure $F(j,x,y)$. This level is composed of two sub-processes, namely **inter-scale clustering** and **inter-orientation fusion**. The inter-scale clustering is denoted as a c within a circle in Figure 3 because this process is actually implemented by correlation, and it is designed to associate the local energy descriptors $\rho(\theta,j,x,y)$ across scales such that the spatial localisation problem is minimised globally. More precisely, for each $\rho(\theta,j,x,y)$, a new feature image $\rho'(\theta,j,x,y)$ is computed by

$$\rho'(\theta,j,x,y) = \begin{cases} \rho_x^j(x-tx1[j], y-ty1[j]) & \theta = 0 \\ \rho_y^j(x-tx2[j], y-ty2[j]) & \theta = \frac{\pi}{2} \\ \rho_{x+y}^j(x-tx3[j], y-ty3[j]) & \theta = \frac{\pi}{4} \\ \rho_{x-y}^j(x-tx4[j], y-ty4[j]) & \theta = \frac{3\pi}{4} \end{cases} \tag{23}$$

where $tx1, ty1$ is a translation vector which is determined by a correlation process such that $\rho'(\theta=0,j,x,y)$ across scales are better aligned in space according to the image structure giving rise to the descriptors. Similarly, the vectors of $tx2, ty2, tx3, ty3$ and $tx4, ty4$ can be determined for the orientations $\pi/2, \pi/4$ and $3\pi/4$, respectively.

Unlike the other levels given above, the inter-orientation fusion is not universal. It is specific to each application and to the meaning of different orientation channels. Currently, a simple formula is used to combine four oriented local energy images into quantitative and orientational measures of local energy denoted as $F(j,x,y)$ and $\alpha(j,x,y)$ which are defined by

$$\begin{aligned} F(j,x,y) &= \sqrt{[\rho'(\theta,j,x,y)|_{\theta=0}]^2 + [\rho'(\theta,j,x,y)|_{\theta=\frac{\pi}{2}}]^2} \\ &\quad + c * \sqrt{[\rho'(\theta,j,x,y)|_{\theta=\frac{\pi}{4}}]^2 + [\rho'(\theta,j,x,y)]^2|_{\theta=\frac{3\pi}{4}}} \end{aligned} \tag{24}$$

$$\alpha(j,x,y) = \arg\left(\frac{\rho'(\theta,j,x,y)|_{\theta=\frac{\pi}{2}}}{\rho'(\theta,j,x,y)|_{\theta=0}}\right) \tag{25}$$

In other words, the total local energy $F(j,x,y)$ is given by a weighted sum of two parts, each part determined by an orthogonal pair of local energy measures. The combined local energy images $F(j,x,y)$ are shown in Figure 4.

The **fourth level** is segmentation, which is carried out in four steps. The first step is filtering $F(j,x,y)$ by Gaussian smoothing, which allows texture density in a local neighbourhood to be computed. The resultant images are then input for clustering at the second step. This exploits the observation that in the feature space, a well-chosen set of features induces well separated clusters corresponding to different classes. Minimisation of the Kullback information distance [14] is applied at each scale and the value of the threshold is determined automatically. One problem of this simple thresholding is that intensity edges are sometimes misclassified as urban textures in our experiments. This is because the Kullback clustering method only uses simple global statistics (i.e. histogram) and limited texture classes (only two classes). Currently, this effect is minimised by applying post-processing. It can be removed by exploiting the fact that the local energy for isolated intensity edges and intensity edge surrounded by many other edges in its neighbourhood (i.e. typical urban texture) behave very differently across scales [26]. The third step is post-processing. Morphological opening and closing is used to remove intensity edges, isolated small patches and holes. Finally, texture boundaries are detected by finding the points of discontinuity in the image.

6.1: Results

The texture segmentation scheme given above is implemented and has been tested on more than 30 real aerial and satellite images. Typical results are shown in Figure 5. Figure 5 (a), (b) show typical IRLS images taken from a low flying aircraft. The goal (part of a system under development for matching images on successive fly-pasts and matching/constructing a map) is to segment rural and urban areas. The patches in Figure 5 (b) correspond to parks within the surrounding urban area. Figure 5 (c) shows the segmentation of a satellite image taken over Plymouth area, the segmentation result is matched quite well with the map over same area. Finally Figure 5 (d) shows the segmentation of two Brodatz textures [7] (cotton canvas and woolen cloth).

7: Conclusions

In this paper, we have introduced an oversampled wavelet multiresolution representation to achieve translation invariance. Then, we developed a complete, decoupled local energy and phase representation of a 2D oversampled wavelet transform. This representation not only provides a better understanding of the wavelet transform in space at each single scale, but also facilitates matching across scales. The usefulness of this decoupled local energy and phase representation is demonstrated by its application segment textures in several classes of natural images.

8: Proofs

Lemma: 5.1 For $k = 1$, Eqn. 12 can also be written by

$$\begin{aligned} DG_1^j(m,n) &= < f(x,y), 2^{2j}G_1(2^j(x-m), 2^j(y-n)) > \\ &= < f(x,y), 2^{2j}\Psi_d(2^j(x-m), 2^j(y-n)) > \\ &- < f(x,y), 2^{2j}H_x\{H_y\{\Psi_d(2^j(x-m), 2^j(y-n))\}\} > \\ &+ i < f(x,y), 2^{2j}H_x\{\Psi_d(2^j(x-m), 2^j(y-n))\} > \\ &+ i < f(x,y), 2^{2j}H_y\{\Psi_d(2^j(x-m), 2^j(y-n))\} > \end{aligned}$$

by noticing $H_y\{H_x\{D_d^j(m,n)\}\} =< f(m,n), 2^{2j}H_x\{H_y\{\Psi_d(2^j(x-m), 2^j(y-n))\}\} >$, we have

$$DG_1^j(m,n) = D_d^j(m,n) - H_y\{H_x\{D_d^j(m,n)\}\} + i(H_x\{D_d^j(m,n)\} + H_y\{D_d^j(m,n)\})$$

To study the orientation selectivity of $DG_1^j(m,n)$, we note that

$$\hat{DG_1^j}(u,v) = \begin{cases} 4\hat{D_d}^j(u,v) & \text{if } u \le \frac{N}{2} \text{ and } v \le \frac{N}{2} \\ 0 & \text{otherwise} \end{cases}$$

where we adopt the convention that $u, v \le \frac{N}{2}$ for positive frequency and $u, v > \frac{N}{2}$ for negative frequency. However, it is known that the distribution of $|\hat{D_d}^j(u,v)|$ is dominant at or close to the $\frac{\pi}{4}$ and $\frac{3\pi}{4}$ axes. Hence the distribution of $|\hat{DG_1^j}(u,v)|$ is only possible to be dominant at or close to $\frac{\pi}{4}$ axis. In other words, $DG_k^1(m,n)$ only gives strong response to spatial structures in or close to the direction[2] $\frac{\pi}{4}$. $DG_3^j(m,n)$ can be proved in a similar

[2] In other words, it gives strong response to spatial structures whose maximum gradient is in, or close to $\frac{3\pi}{4}$.

way.

□

Theorem: 5.1

1. Using Eqn. 14, 15, we have

$$\begin{aligned} Re\{DG_1^j(m,n)\} &= D_d^j(m,n) - H_y\{H_x\{D_d^j(m,n)\}\} \\ Re\{DG_3^j(m,n)\} &= D_d^j(m,n) + H_y\{H_x\{D_d^j(m,n)\}\} \end{aligned}$$

Adding them together,

$$\begin{aligned} D_d^j(m,n) &= \frac{1}{2}(Re\{DG_1^j(m,n)\} + \{DG_3^j(m,n)\}) \\ &= \frac{1}{2}(\rho_{x+y}\cos(\varphi_{x+y}) + \rho_{x-y}\cos(\varphi_{x-y})) \end{aligned}$$

2. For each scale j, applying Parseval's theorem, we have

$$\begin{aligned} \sum_m \sum_n |D_d^j(m,n)|^2 &= \sum_u \sum_v |\hat{D}_d{}^j(u,v)|^2 \\ &= \sum_u \sum_v |\hat{f}(u,v)\overline{\hat{\Psi}}_d(2^{-j}u, 2^{-j}v)|^2 \\ &= \sum_u \sum_v |\hat{f}(u,v)|^2 |\hat{\Psi}_d(2^{-j}u, 2^{-j}v)|^2 \end{aligned} \tag{26}$$

Considering the Fourier transform of the wavelet Ψ_d, we have

$$\begin{aligned} |\hat{\Psi}_d(2^{-j}u, 2^{-j}v)|^2 &= \frac{1}{16}|\sum_{k=1}^{4} \hat{G}_k(2^{-j}u, 2^{-j}v)|^2 \\ &= \frac{1}{16}(\sum_{k=1}^{2} |\hat{G}_k(2^{-j}u, 2^{-j}v)|^2 + \sum_{k=3}^{4} |\hat{G}_k(2^{-j}u, 2^{-j}v)|^2 \\ &= \frac{1}{8}(|G_1(2^{-j}u, 2^{-j}v)|^2 + |G_3(2^{-j}u, 2^{-j}v)|^2) \end{aligned} \tag{27}$$

Substituting Eqn. 27 to Eqn. 26,then

$$\begin{aligned} \sum_m \sum_n |D_d^j(m,n)|^2 &= \frac{1}{8}\sum_u \sum_v |\hat{f}(u,v)\hat{G}_1(2^{-j}u, 2^{-j}v)|^2 + |\hat{f}(u,v)\hat{G}_3(2^{-j}u, 2^{-j}v)|^2) \\ &= \frac{1}{8}\sum_m \sum_n (|DG_1^j(m,n)|^2 + |DG_3^j(m,n)|^2) \\ &= \frac{1}{8}\sum_m \sum_n ((\rho_{x+y}^j(m,n))^2 + (\rho_{x-y}^j(m,n))^2) \end{aligned}$$

□

9: Acknowledgement

This work was supported by a strategic research programme of the Defence Research Agency. We acknowledge many interactions with Margaret Varga, Paul Ducksbury, Phil

Kent, and John Radford. Michael Brady thanks INRIA for hosting his sabbatical, in particular Nicholas Ayache, Marc Berthod, Oliver Faugeras, and Zhengyou Zhang. Thanks also to Robyn Owens and Andrew Zisserman for comments on the ideas presented here. We really acknowledge the inspiration we have had from Azriel Rosenfeld's work over the past quarter century.

References

[1] N. Ahuja and A. Rosenfeld. Mosaic models for textures. *IEEE PAMI*, PAMI-3:1–11, 1981.

[2] R. Bajcsy and L. Lieberman. Computer identification of visual surfaces. *Computer Graphics and Image Processing*, pages 118–130, 1973.

[3] J. Beck. Similarity grouping and peripheral discriminability under uncertainty. *Am. J. Psychol.*, pages 1–19, 1972.

[4] J. Beck et al., editors. *Organization and Representation in Perception, ch. Textural Segmentation*. Erlbaum, Hillsdale, NJ, 1982.

[5] J. Beck, J. Hope, and B. Rosenfeld. *Human and Machine Vision*. Academic Press, 1983.

[6] A. C. Bovik, M. Clark, and W. S. Giesler. Multichannel texture analysis using localized spatial filters. *IEEE PAMI*, 12:55–73, 1990.

[7] Phil Brodatz. *Textures*. Dover Publications, Mineola NY, 1966.

[8] P. H. Carter. Texture discrimination using wavelets. In *SPIE applications of digital image processing XIV*, volume 1567, pages 432–438, 1991.

[9] R. Chellappa and S. Chatterjee. Classification of textures using gaussian markov random fields. *IEEE Trans. on Acoustic Speech Signal Processing*, 33(4):959–963, 1985.

[10] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Am.*, 2:1160–1169, 1985.

[11] H. Derin and H. Elliot. Modeling and segmentation of noisy and textured images using gibbs random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9(1):39–55, 1987.

[12] D. Gabor. Theory of communication. *Journal of IEE*, 93:429–457, 1946.

[13] Robert M. Haralick, K.Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE SMC*, SMC-3(6):610–621, 1973.

[14] Robert M. Haralick and Linda Shapiro. *Computer and Robot Vision*. Addison-Wesley Publishing Company, 1992.

[15] A. K. Jain and F. Farrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24:1167–1186, 1991.

[16] A. Laine and J. Fan. Texture classification by wavelet packet signatures. *IEEE PAMI*, 15(11):1186–1190, 1993.

[17] J. Malik and P. Perona. A computational model for texture segmentation. In *IEEE Proceeding of Computer Vision and Pattern Recognition*, 1989.

[18] Jitendra Malik and Pietro Perona. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America A*, 7(5):923–932, 1990.

[19] Stephane G. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE ASSP*, 37:2091–2110, 1989.

[20] B. S. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random field models. *PAMI*, 13(5):478–482, 1991.

[21] Uwe Mussigmann. Homogeneous fractals and their application in texture analysis. *Fractals in the Fundamental and Applied Sciences, edited by H.-O. Peitgen, J.M. Henriques and L.F. Penedo*, pages 269–281, 1991.

[22] Alex P. Pentland. Fractal-based description of natural scenes. *IEEE PAMI*, PAMI-6(6):661–674, 1984.

[23] A. Rosenfeld and A. C. Kak. *Digital Picture Processing*. Academic Press, 1982.

[24] A. Rosenfeld and E. Troy. Visual texture analysis. *Technical Report 70-116, University of Maryland, College Park*, 1970.

[25] J.S. Weszka, C. R. Deya, and A. Rosenfeld. A comparative study of texture measures for terrain classification. *IEEE Trans. on Systems, Man and Cybernetics*, SMC-6:269–285, 1976.

[26] Z. Xie. *Multi-scale Analysis and Texture Segmentation*. PhD thesis, University of Oxford, 1994.

[27] Zhi-Yan Xie and Michael Brady. Fractal dimension image for texture segmentation. In *Proceedings of 2nd International Conference on Automation, Robotics and Computer Vision*, volume 1, pages CV–4.3.1 to CV–4.3.5, 1992.

[28] Zhi-Yan Xie and Michael Brady. A decoupled local energy and phase representation of a wavelet transform. In *VCIP'95 (Visual Communications and Image Processing)*, 1995.

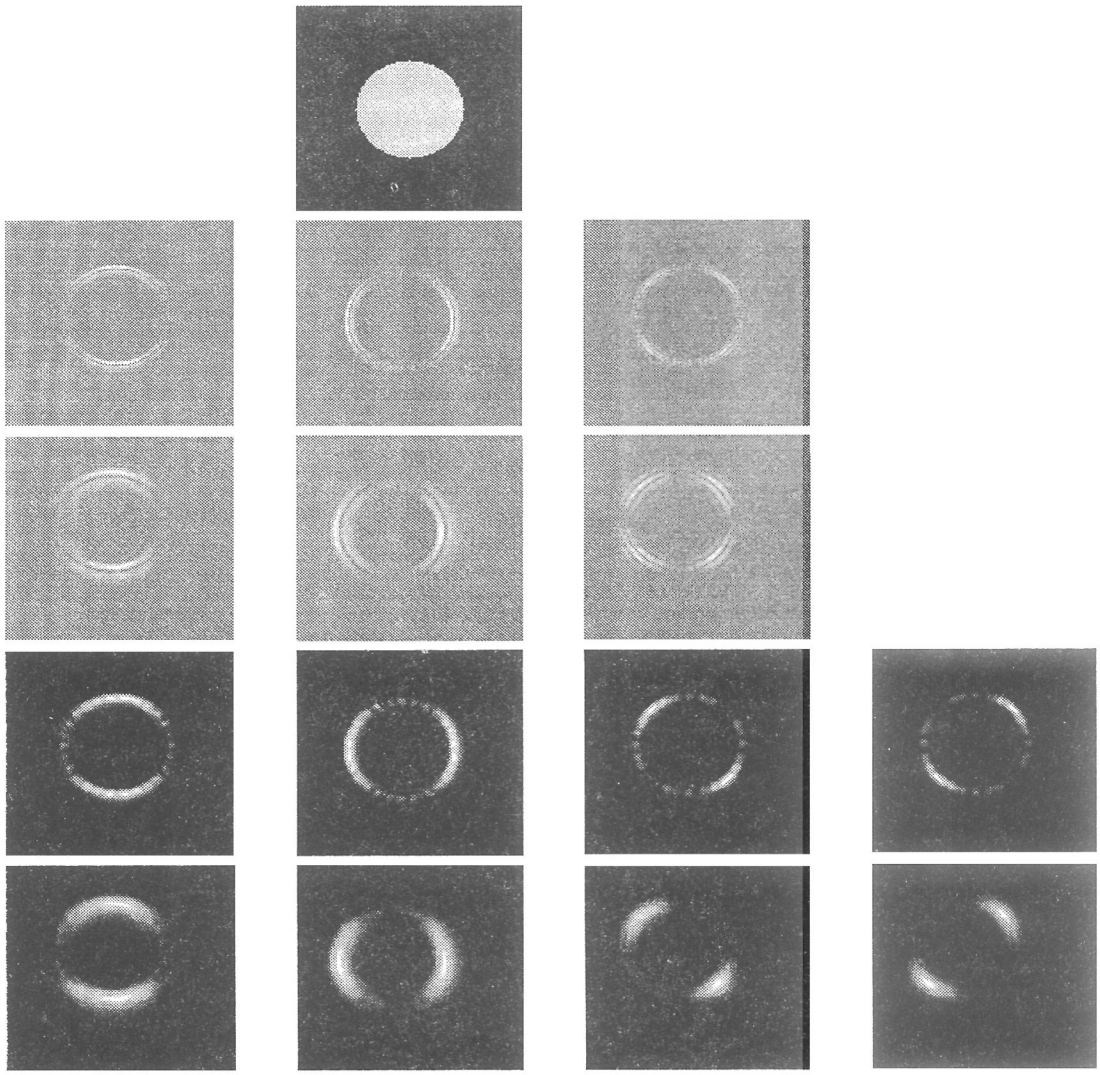


Figure 1. The wavelet decompositions of a synthetic image. top to bottom (from left to right): original synthetic image; the detail images at $j = -2, j = -3$; the local energy images at scale $j = -2, -3$, respectively. The Daubechies wavelet of length 4 is used throughout.
ordinary paragraph.

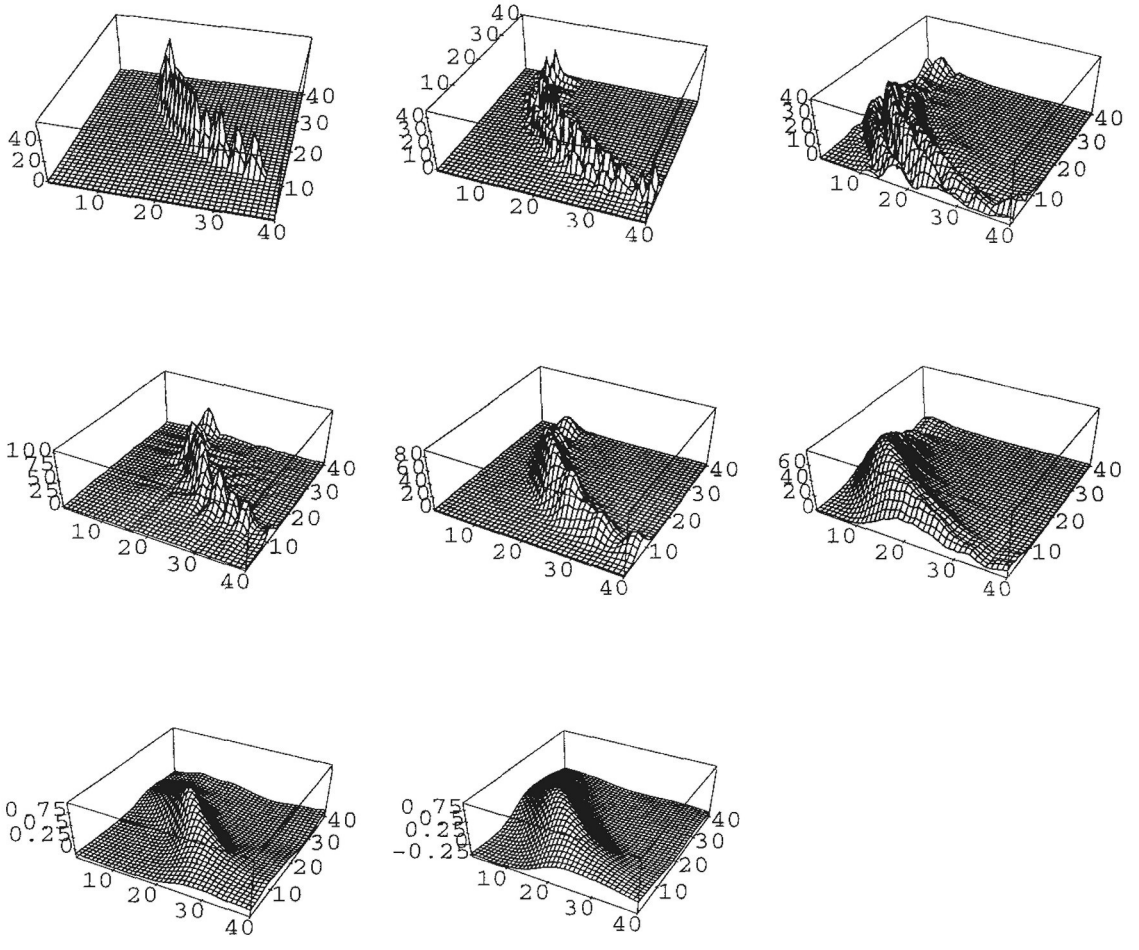


Figure 2. Comparison between the modulus and the local energy of the wavelet transform. Top row (left to right): the plot of patches of $|D_d^j|$ of the synthetic image at scales $j = -1, -2, -3$, respectively; middle row: the plot of the local energy ρ_{x-y}^j for the same patch at scale $j = -1, -2, -3$, respectively; bottom row: the plot of the linear correlation coefficients of two local energy images at adjacent scales, $j = -1, -2$ and $j = -2, -3$ respectively.
ordinary paragraph.

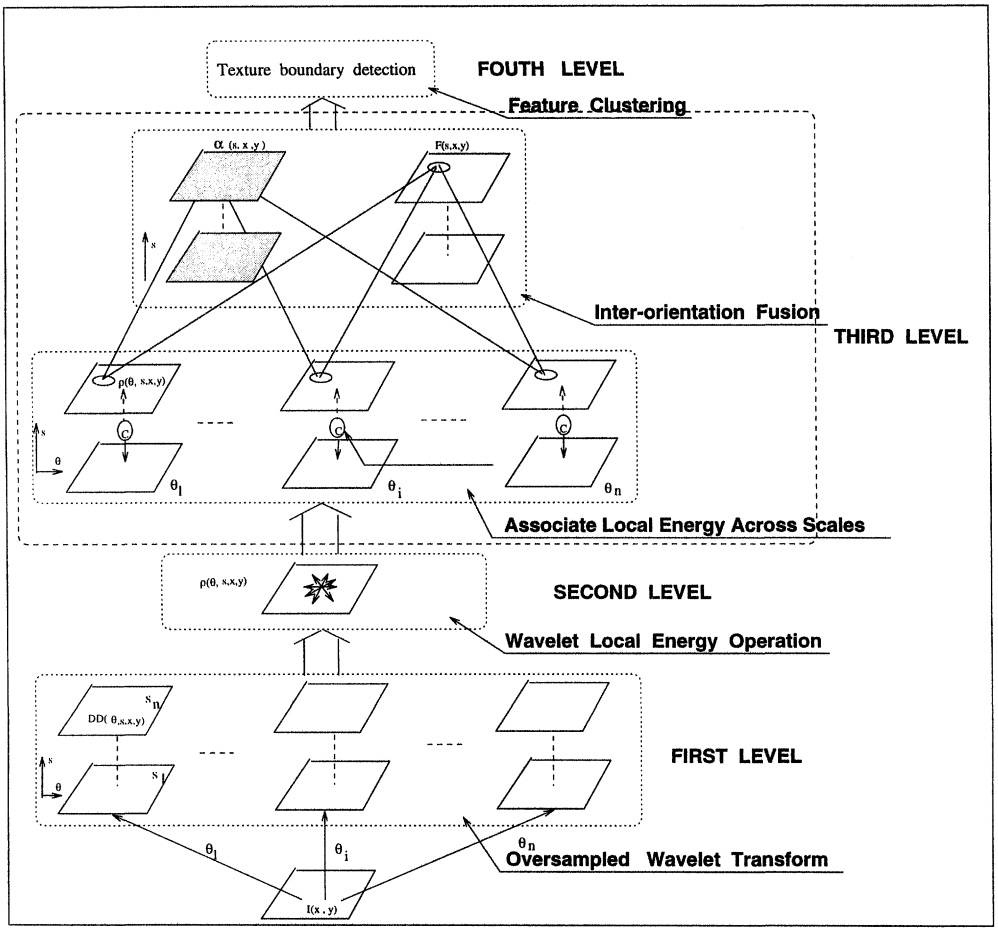


Figure 3. A schematic view of texture segmentation system in wavelet scale-space. The 1st level is to represent an image by a set of wavelet detail images $D(\theta, s, x, y)$, where each column and row is corresponding to the wavelet representation at different orientation and scale, respectively; the 2nd level is to derive a set of local energy images $\rho(\theta, s, x, y)$ by applying the local energy operation to each of the images; the 3rd level is to derive two feature images $F(s, x, y)$ and $\alpha(s, x, y)$ at different scales; the 4th level is to detect texture boundaries.
ordinary paragraph.

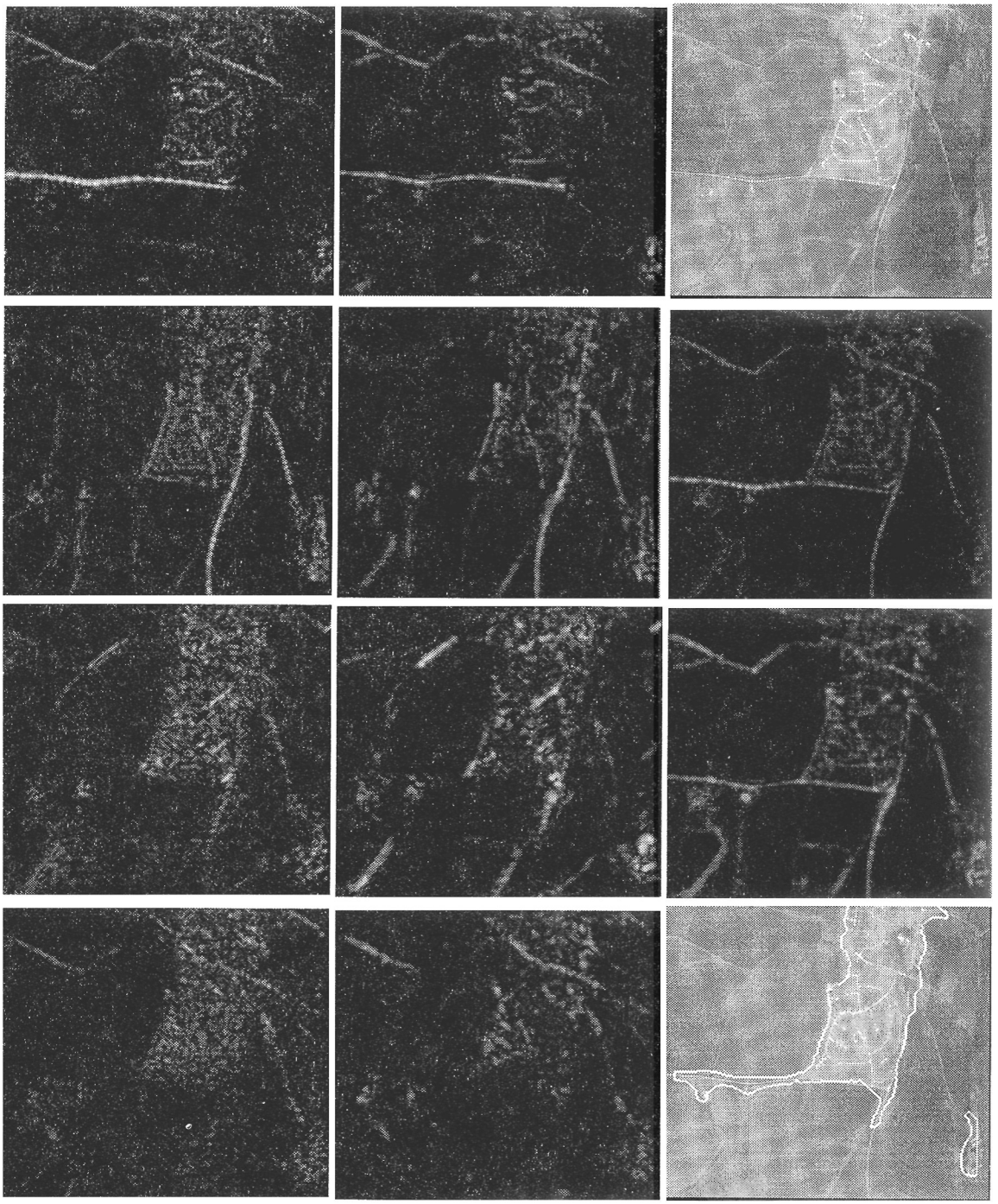
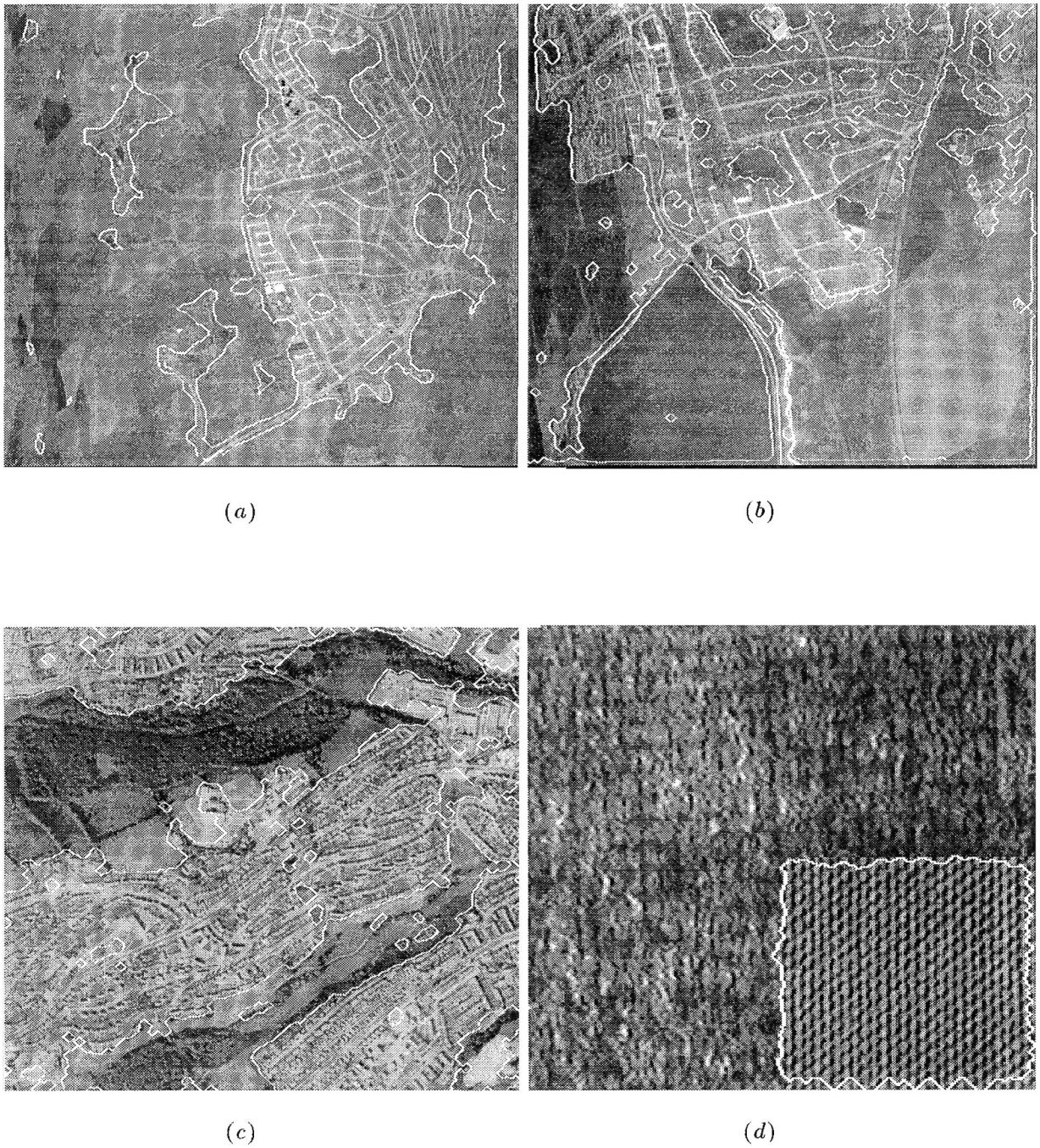


Figure 4. The left (middle) column from top to bottom: the local energy images $\rho(\theta, s, x, y)$ at scale $j = -1$ ($j = -2$) in horizontal, vertical, $\frac{\pi}{4}$ and $\frac{3\pi}{4}$ directions; the right column (top to bottom): original image 256×256, the combined local energy image $F(s, x, y)$ at scale $s = 2^{-j}, j = -1, -2$, respectively, where $c = 3.0$ and the segmentation result.
ordinary paragraph.



(a) (b)

(c) (d)

Figure 5. Examples of texture segmentation results. Texture boundaries are extracted and superimposed on their original images. Top row: Urban regions have been extracted for real IRLS aerial images; bottom row (left to right): Urban regions have been extracted for a satellite image, cotton canvas (in right bottom corner) have been picked up from woolen cloth background for a Brodatz montage image. ordinary paragraph.

A Transform For Multiscale Image Segmentation

Narendra Ahuja
Beckman Institute and Coordinated Science Laboratory
University of Illinois at Urbana-Champaign

Abstract

This paper describes a new transform to extract image regions at all geometric and photometric scales. It is argued that linear approaches such as convolution and matching have the fundamental shortcoming that they require *a priori* models of edge geometry. The proposed transform avoids this limitation by letting the structure emerge, bottom-up, from interactions among pixels, in analogy with statistical mechanics and particle physics. The transform involves global computations on pairs of pixels followed by vector integration of the results, rather than scalar and local linear processing. An attraction force field is computed over the image in which pixels belonging to the same region are mutually attracted and the region is characterized by a convergent flow. It is shown that the transform possesses properties that allow multiscale segmentation, or extraction of original, unblurred structure at all different geometric and photometric scales present in the image. This is in contrast with much of the previous work wherein multiscale structure is viewed as the smoothed structure in a multiscale decomposition of image signal. Scale is an integral parameter of the force computation, and the number and values of scale parameters associated with the image can be estimated automatically. Regions are detected at all *a priori* unknown scales resulting in automatic construction of a segmentation hierarchy, in which each pixel is annotated with descriptions of all the regions it belongs to. Although some of the analytical properties of the transform are presented for piecewise constant images, it is shown that the results hold for more general images, e.g., those containing noise and shading. Thus the proposed method is intended as a general approach to multiscale, integrated edge and region detection, or low-level image segmentation.

1 Introduction

This paper is concerned with the problem of low level image segmentation, or partitioning of an image into homogeneous regions, that represent low level image structure. A region can be characterized as possessing a certain degree of interior homogeneity and a contrast with the surround which is large compared to the interior variation. This is a satisfactory characterization from both perceptual and quantitative viewpoints. The type of homogeneity (constancy, smoothness) and the magnitude of the contrast may vary, and the regions may have arbitrary size and shape. Different values of allowed homogeneity and contrast lead to different partitioning of image into regions. As greater homogeneity and lower contrast values are allowed, new, smaller regions emerge in the partition. Thus, a decrease in the minimum acceptable contrast leads to increased, hierarchical decomposition of the image, which culminates in constant-value regions at the bottom of the hierarchy. The depth of the hierarchy and the values of region homogeneity, contrast, shape and size parameters associated with the different levels vary across the image. Features such as

0-8186-7644-2 $5.00

depth and branching factor of the tree thus defined are unrelated across subtrees, each solely determined by the image and therefore *a priori* unknown. The homogeneity and contrast parameters associated with different image regions will be said to form the set of photometric scales present in the image, while the region shapes and sizes will be said to define the geometric scales present.

Finding a solution of the low level segmentation problem poses two main challenges. First, a valid image region must be detected regardless of its shape, size, degree of homogeneity, and contrast. Second, all geometric and photometric scales at which regions happen to occur across an image must be identified. If these two problems are solved, the result will be a segmentation tree representing the multiscale, low level, image structure. To obtain such a tree for an arbitrary image using intensity based homogeneity and contrast is the objective of image segmentation pursued in this paper.

Limited work has been done to meet both of the above challenges. Much of the previous work on multiscale analysis is concerned with a scale-space decomposition of the image signal, determined by a single scale parameter. The decomposition amounts to a blurring of the image to different degrees. The image structure is present across this scale-space continuum and the extraction of image regions of different sizes and contrasts from this continuum is not addressed. Further, even if they were extracted, the regions in the different decompositions would be correspondingly smoothed. Automatic estimation of scale parameters is typically not addressed. Even at a given scale, robust detection of a region continues to be an area of active investigation, mainly through the work on edge detection. Region detection such that the detected boundaries are closed and coincident with the true region boundaries regardless of region parameters is not a solved problem. Most methods are linear and often use restrictive region models, e.g., allowed geometric and photometric complexity of edges. Although these models simplify processing, they cause fundamental limitations in the detection accuracy and sensitivity achieved which is partly why the problem of region and edge detection continues to evade a satisfactory solution.

A central theme of this paper is to achieve both of the above goals of accurate region detection and automatic scale estimation. This is accomplished by introducing a new transform which converts an image into another image having two major properties. First, for a specific pair of scale values, the transform leads to well-defined signatures of corresponding image regions which are easy to detect. The transform definition incorporates the duality of interior- and edge-based descriptions of regions. Thus, the transform performs integrated edge and region detection, and can be viewed as a multiscale blob and edge detector at the same time. Second, all scale values for which regions occur in the image can be computed by analyzing the results of the transform as the scale values used are varied.

A major motivation for the transform comes from a desire to develop a segmentation method which is not rigidly bound to specific models of regions, e.g., models of spatial variation. It is desired that groupings of pixels that reflect a smooth photometric variation and stand in relative contrast to their surround be detected as regions, regardless of the exact nature of variation and contrast values. This appears necessary to deal with the variety characteristic of real imagery. To achieve such performance, the transform computes affinities among image *points* or *pixels* for grouping with other pixels, letting the structure emerge bottom-up from "interactions" among the pixels instead of imposing *a priori* chosen models of region edges and interior. As one consequence of this, the emergent region geometry is not restricted, since pixels can group together to form any connected set.

The transform computes a family of force fields for a given image where the force vector at a point denotes its affinity to the rest of the image. On either side of a region boundary, the pixels have high affinities, but there is little affinity between pixels across regions.

The strength of interaction between pixels, and consequently their affinities, depend upon their distances and contrasts, and this allows association of the computed affinities and segmentation with spatial and photometric scales. Since the transform allows interaction between a pixel and all other pixels, it can be viewed as collecting globally distributed evidence for image structure and making it available locally, e.g. at the locations of region edges and medial axes (The medial axis of a region is defined as the locus of points inside the region which are equidistant from two or more points on the region boundary [6, 17]). The regions are encoded in the force field via distinct signatures amenable to robust, local identification. In this sense, the transform performs Gestalt analysis.

This paper introduces the transform and shows how it can be used for segmentation (the basic idea of the transform can be found in [1]). It does not present specific segmentation algorithms. The segmentation is intended to represent low-level image structure at all scales, thus with applicability to textured as well as smooth images. To analyze and illustrate the basic properties of the transform, we model regions, whenever necessary, as possessing uniform gray levels and step edges. However, the transform properties and segmentation results are shown to apply to images containing general types of regions as discussed above, e.g., having shading and noise. Section 2 discusses some basic desired characteristics of segmentation and how they motivate the proposed approach. Section 3 describes the transform, describes some of its properties of interest, and shows how these properties facilitate multiscale segmentation. Section 4 analyzes the segmentation performance of the transform. Section 5 presents concluding remarks.

2 Background and Objectives

In this section, we first discuss past work on the two major subproblems of image segmentation: structure detection at a single scale and multiscale analysis (Sec. 2.1). This leads us to formulate the characteristics desired in a satisfactory segmentation (Sec. 2.2).

2.1 Two Aspects of Image Segmentation

We will first review the past work on photometric and geometric models of a region used for segmentation at a single scale. [26] estimates a 2D functional that minimizes a cost function comprised of the difference between the images and estimated intensity values, the length of detected edges, and the variation in the functional away from edges, which are combined using *a priori* chosen relative weights. Morphological methods are used in [23] to detect regions as intensity hills in grayscale landscape. Although, a region can be detected by identifying either its interior pixels or edges, the latter method has been investigated more extensively. An edge separates two different regions and thus two different types of gray level populations. Edge detection methods use different models of edge geometry, and gray level variation along edge as well as within region. These models are fitted to local pixel populations to determine if an edge is present or not. Such local responses are then combined to derive a more global segmentation. Clearly, the validity of the models of the edge as well the gray level populations are critical factors in achieving a valid segmentation. We will now review some models used in the previous work. It is common to treat the problem of edge detection as mainly that of selecting a point along the intensity profile across edge, assuming such a profile can be extracted from the image. Accordingly, a model of the intensity profile is used to precisely define an edge and to optimally detect its location. Different types of intensity models of an edge have been proposed, according to the nature of the two populations and the spatial profile of the

transition from one to the other across the edge [30, 29, 5, 11, 14]. To meet the assumption that edge profile through a pixel can be identified, it is common in edge detection work to implicitly or explicitly use a model of edge curvature. The use of such geometric model constrains the number of possible, different subdivisions of the pixel neighborhood (e.g., a 3x3 or a 5x5) into two regions which must be analyzed to detect the presence of an edge in the neighborhood. For example, the assumption of local straightness of edge is common which makes it very easy to select neighborhoods on the two sides of the edge. [22] assumes that the edge is locally straight (and that the intensity changes linearly along a direction parallel to the edge.) Nalwa and Binford [27] assume straightness to extract a sample edge profile. Even the computation of gradient which is common to many edge detectors, e.g., the diffusion based methods [21, 34],romeney), implicitly assumes local edge straightness. The same can be said about Laplacian based edge detectors. The use of straightness is very explicit in the different types of discrete edge masks each of which is meant to detect a different edge orientation [31]. To detect intensity facets meeting at an edge [13], a model of edge geometry is required so candidate neighborhoods from each side of the edge can be identified. The work on optimal edge detection (e.g., [9]) is also subject to the validity of the assumed model of the edge geometry. In short, the work on edge detection has lead to different approaches to estimate edge location and orientation for edges having some (implicitly or explicitly) assumed local curvature properties. Image edges do not always conform to these assumptions, and deviations lead to detection errors. Examples of such errors incurred using the Laplacian-of-Gaussian operator for different edge geometries can be found in [4]. To avoid some of these problems, [12] uses the Markov random field model to obtain an estimate similar to that in [26] but allows for end points, corners and junctions in the edge models used. Another example of an approach that avoids the dependence on geometric models of edge is given in [28] where interpixel correlations in spatio-temporal space are considered instead of interwindow correlations.

The second major aspect of segmentation is related to scale. As we stated earlier, scale as pursued in this paper is associated with both geometric and photometric sensitivity to detail. Thus, a pixel may simultaneously belong to different regions each having a different contrast value and size, giving rise to the tree representation mentioned earlier. Large regions may be said to have a coarse spatial scale while smaller sizes may be said to be associated with finer spatial scales. Analogously, an edge contour which separates two regions of a given contrast scale may not be detected at a higher scale associated with a larger contrast. The exact number and parameters of scales for a given image are *a priori* unknown. Therefore, multiscale segmentation must automatically estimate these parameters and detect the corresponding regions. Although the general notion of multiscale operators has been examined for a long time [32], there has been limited work on definition, analysis and automatic estimation of multiscale image structure as pursued in this paper. Our objective here is to separate *original* (unsmoothed) image structure at different scales (regions with different sizes and contrasts), as well as identify the spatial and topological relationships among the regions. We obtain a multiscale structural decomposition of the image, and not a multiscale decomposition of the image signal as performed in much of the past work [18, 35, 36, 20] where coarse scale structure is detected from blurred images and is therefore a smoothd version of the original image structure. In addition to accruing edge displacement error [4], the latter leads to artifcats due to multiscale analysis, e.g., phantom edges [10]. Among other approaches to multiscale segmentation, multiscale blob detection using morphological methods is described in [7, 19], and computation of multiscale medial axis representation is discussed in [25, 3, 17].

2.2 Desired Characteristics and Objectives

The above discussion leads us to the following desired characteristics of multiscale structure detection and segmentation.

A. Shape and Topology Invariance: The regions should be correctly detected regardless of their shapes and relative placement. For example, an edge point must be detected at only one and the correct location, regardless of whether the edge in the vicinity of the point is straight, curved or even contains a corner or a vertex where multiple regions meet.
B. Photometric Scaling: It should be possible to detect all regions which are in contrast to their surround, regardless of the actual degree of within-region homogeneity and the value of the contrast. Regions having large contrast may be associated with higher scales.
C. Spatial Scaling: It should be possible to detect all regions regardless of their shapes and sizes. Higher scales may be associated with larger regions.
D. Stability and Automatic Scale Selection: Image structures associated with different scales correspond to segmentations that are locally invariant to changes in geometric and contrast sensitivities. Since the contrasts and sizes of regions contained in an arbitrary image are *a priori* unknown, they should be identified automatically.

The transform presented in this paper has been motivated by the objective of achieving these desired characteristics. Specifically, the objective is to derive multiscale segmentation of the image and represent it through a hierarchical, tree structure in which the different image segments, their parameters, and their spatial interrelationships are made explicit. The bottom (leaf) nodes of the hierarchy correspond to regions consisting of individual pixels or connected components of constant gray level, and the path from a leaf to the root node specifies how the leaf regions recursively merge with adjacent regions to form larger regions each of which is homogeneous with respect to its surround and is characterized by its own contrast. Alternate representations of the same image structure and contrast information are also possible, e.g., by ordering regions according to contrast. In this paper, we will not dwell on the different possible data structures that could be used for representation. Rather, we will demonstrate how the transform extracts information about the *a priori* unknown region geometries, homogeneities and contrasts, and associates this structural information with each image pixel. Any specific image representation may be constructed from such annotated pixel array.

3 The Transform and Image Segmentation

In this section we first discuss how the problems with the previous methods and the desired characteristics of the segmentation motivate an approach such as that underlies the proposed transform (Sec. 3.1). We then introduce the transform (Sec. 3.2) and present its properties (Sec. 3.3) that demonstrate how the transform makes explicit the image structure and facilitates image segmentation, and why the resulting segmentation possesses the desired characteristics listed in Section 2.2. Section 3.4 describes how a given image region appears in the vector field computed by the transform. In Sec. 3.5, we discuss the estimation of the unknown scale parameters associated with an image, which are required to extract the unknown structures present in different parts of the image. Sec. 3.6 describes the hierarchical image structure that is extracted as the final result of segmentation. Whenever necessary, we will consider constant-value regions to analytically and qualitatively describe the transform behavior. However, the properties of the transform responsible for its segmentation capability remain valid for images containing more general

types of regions, such as those having noise (statistical constancy) and shading (smoothly varying values or higher-order homogeneity) as will be explained in Sec. 4. Thus, the transform is proposed for use in segmentation of piecewise smooth images.

3.1 Overcoming Limitations of Linear Processing via the Transform

In the literature, different models of edge profile (step, ramp, roof) and their validity have been investigated. However, the limitations and impact of the assumptions made about edge geometry have received limited attention. Since any convolution kernel for edge detection must incorporate a template for the expected edge geometry, no linear, convolution based approach would avoid the limitations resulting from the use of geometric models of edge. In the digital case, one could attempt to circumvent the problem by enumerating all possible edge geometries in a neighborhood. But the number of resulting kernels will fast increase with neighborhood size and will be prohibitively large for any reasonable size neighborhood.

The inspiration for the proposed solution comes from physics where microscopic homogeneity of physical properties leads to islands of, say, similar particles or molecules. An island shape is congruent with the space occupied by a set of contiguous, similar particles, whatever the complexity of the boundary! The particles group together and coalesce into regions based on the similarity of their intrinsic properties only, regardless of their relative locations. The common property of particles then characterizes the region they form. As an alternate analogy, the grouping process is like the alignment of microscopic domains over an area of ferromagnetic material. The key process is that of interaction among particles which leads to bindings among similar particles.

The problem of segmentation has similarities to the above physical process. The goal is to find a partition of the image, regardless of the boundary complexity, such that each cell of the partition has a characteristic property, say, homogeneity of gray level. This analogy suggests a formulation of the segmentation process in terms of a suitably defined method of interpoint interaction - one that would group, bottom-up, each set of points of the same property to form a region having a boundary of any complexity. Being a parameter of the grouping process, different acceptable degrees of the presence of the property within a region would yield groupings over different regions, making scale an integral part of structure detection.

In the next section, we introduce a transform which achieves the above grouping.

3.2 The Transform

The transform converts the image I into a vector field $\mathbf{F}$. The vector $\mathbf{F_p}$ at an image location $\mathbf{p}$ is defined as

$$\mathbf{F_p} = \int_{\mathbf{q} \neq \mathbf{p}} d_s(\mathbf{r_{pq}}, \sigma_s(\mathbf{p})) d_g(\Delta I, \sigma_g(\mathbf{p})) \hat{\mathbf{r}}_{\mathbf{pq}} \mathbf{dq} \tag{1}$$

where
$\hat{\mathbf{r}}_{\mathbf{pq}}$ = unit vector in the direction from $\mathbf{p}$ to another image location $\mathbf{q}$;
$\sigma_s(\mathbf{p})$ = spatial scale parameter at $\mathbf{p}$; related to the shortest distance to region boundary; all valid $\sigma_s(\mathbf{p})$ values are computed automatically;
$\sigma_g(\mathbf{p})$ = photometric scale parameter at $\mathbf{p}$; denotes contrast of region with surround; all valid $\sigma_g(\mathbf{p})$ values are computed automatically;
ΔI = absolute gray level difference between image points under consideration;
$d_s(\mathbf{a}, b)$ = A nonnegative, nonincreasing function of $\|\mathbf{a}\|$, not identically 0 for $\|\mathbf{a}\| \leq b$, and 0 for $\|\mathbf{a}\| > b$, and

$d_g(a,b)$ = A nonnegative, nonincreasing and symmetric function of a, not identically 0 for $a \leq b$, and 0 for $a > b$.

Since $d_g(a,b)$ as defined above cannot be a linear function of a for unrestricted values of a, the transform is a function of image input.

3.2.1 Formulation of the Transform

Having stated the definition of the transform, we will now explain the motivation behind this formulation, and construct the definition from basic principles.

Consider a region R and a point P inside it. It is desired that the similarity of P to all the other points within R, relative to those outside R, be recognized, regardless of the distance to and curvature of the nearest region boundary. To do this, the transform defines a neighborhood around P which is sufficiently small so that the pixels within the neighborhood which have the most influence on the computation at P are within R. The similarity is estimated by comparing the gray level at P with those of the points in the neighborhood, rather than testing some (position independent) gray level statistics of points withing the neighborhood, or comparing the statistics of different sets of pixels near P. An estimate of the local shape of R near P, specifically of the nearest region border, is computed. This is achieved by a vector integration of the results of pairwise comparisons of points at different orientations, instead of computing a scalar, weighted average at P as is done in linear methods. Contiguous points having compatible estimates of local shape of R are grouped together. This results in the detection of an arbitrarily large region of arbitrary shape.

To capture the local region geometry, the transform computes an attraction-force field over the image wherein the force at each point denotes its affinity to the rest of the image. The force vector points in the direction in which the point experiences a net attraction from the points in the rest of the image. For example, a point inside a region would experience a force towards the interior of the region. This force is computed as the resultant of attraction-forces due to all other image points. If $F(\mathbf{p},\mathbf{q})$ denotes the magnitude of the force vector $\mathbf{F}(\mathbf{p},\mathbf{q})$ with which a pixel P at location $\mathbf{p}$ is attracted by another pixel Q at location $\mathbf{q}$, then the transform is given by

$$\mathbf{F}(\mathbf{p},\mathbf{q}) = F(\mathbf{p},\mathbf{q})\hat{\mathbf{r}}_{\mathbf{pq}},$$

where $\hat{\mathbf{r}}_{\mathbf{pq}}$ denotes the unit vector in the direction from P to Q, i.e.,

$$\hat{\mathbf{r}}_{\mathbf{pq}} = \frac{\mathbf{q}-\mathbf{p}}{\|\mathbf{q}-\mathbf{p}\|}$$

In the real image plane, an image is transformed into a continuous vector field. The vector $\mathbf{F}_\mathbf{p}$ at point P is given by

$$\mathbf{F}_\mathbf{p} = \int_{\mathbf{q}\neq\mathbf{p}} F(\mathbf{p},\mathbf{q})\hat{\mathbf{r}}_{\mathbf{pq}}\mathbf{dq} \tag{2}$$

where $\mathbf{q}$ can be any image location other than $\mathbf{p}$. In the discrete case,

$$\mathbf{F}_\mathbf{p} = \sum_{\mathbf{q}\neq\mathbf{p}} F(\mathbf{p},\mathbf{q})\hat{\mathbf{r}}_{\mathbf{pq}}$$

We need to specify what forms the force function $F(\mathbf{p},\mathbf{q})$ could take. We will do so by identifying the characteristics that any such function must possess, to yield the correct segmentation for a given pair of spatial and photometric scales as well as exhibit appropriate

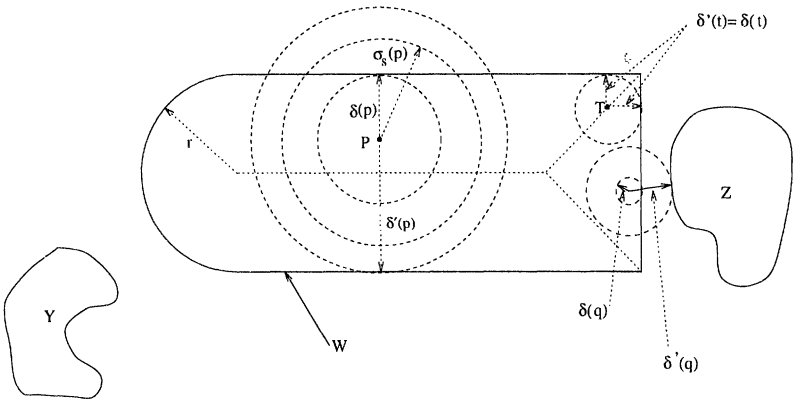


Figure 1. The medial axis of the region W is the locus of points equidistant from multiple points on W's border, shown here by dotted lines. A point P's distance to the nearest border point is given by $\delta(\mathrm{p})$. A disk D of radius $> \delta'$ may intersect with W in more than one connected components (e.g., point P), or it may intersect with other regions as well (e.g., point Q). For points along the medial axis, $\delta' = \delta$ (e.g., point T). The spatial scale parameter σ_s at any point is defined as a value between δ and δ' (radius of the middle circle for point P).

behavior across multiple scales. Specific choices of force functions having these characteristics will then define different instances of the transform.

Since the presence of an edge of a region at any given spatial scale must be determined by its adjoining regions rather than by distant points across other intervening regions, the force F exerted on a given pixel P by another pixel Q should be a nonincreasing function of the distance between P and Q. Further, a pixel should be attracted more to a pixel within its own region than to one in a different region, and the attraction between two pixels should depend only on the magnitude of their gray level difference and not its sign. This is accomplished by making F to be a nonincreasing and symmetric function of the difference between the gray levels of P and Q.

Since both spatial and photometric vicinities are relative to the scales of interest, let us now consider how to integrate the scale information in the computation of F. Consider a region W which exists amidst many other regions in an image (Fig. 1). To detect any structural characteristics of W, the computation of F at a point must not extend to nonlocal parts of W or to other regions, since otherwise the result will depend upon nonlocal structure of W or multiregion structure. To be specific, first consider the situation where a point P is inside W but not on its medial axis. Let $\delta(\mathbf{p})$ denote the distance to the unique border point (or border segment) of W closest to P, and let D denote a disk of radius $r = \delta(\mathbf{p})$ centered at P. If r is increased, for some value $r = \delta'(\mathbf{p}) > \delta(\mathbf{p})$, D's intersection with W will begin to consist of multiple disconnected regions or D will begin to intersect with other regions near W (e.g., regions Y and Z in Fig. 1). Now consider the second situation where the point P is along the medial axis of W. For any such point, $\delta'(\mathbf{p}) = \delta(\mathbf{p})$ since a disk of radius $> \delta(\mathbf{p})$ will intersect W in multiple disconnected components, and possibly other regions. In order that $\mathbf{F_p}$ reflects the structure of W surrounding P, it must be ensured that the contribution to F at $\mathbf{p}$ from points at a distance $\geq \delta'(\mathbf{p})$ is negligible. Such spatial

locality of computation is enforced by including a spatial (or geometric) scale parameter σ_s in the definition of F. $\sigma_s(\mathbf{p})$ associates a "cut-off distance" with $\mathbf{p}$ such that the points farther than this distance make negligible contribution to F value at $\mathbf{p}$. In particular, we define the spatial scale parameter $\sigma_s(\mathbf{p})$ at each point $\mathbf{p}$ in some region R as having a value given by $\sigma_s(\mathbf{p}) = \delta(\mathbf{p}) + \alpha(\delta'(\mathbf{p}) - \delta(\mathbf{p}))$, where $0 < \alpha < 1$. Thus, $\delta(\mathbf{p}) < \sigma_s(\mathbf{p}) < \delta'(\mathbf{p})$ at all points inside R except along the medial axis where $\sigma_s(\mathbf{p}) = \delta(\mathbf{p}) = \delta'(\mathbf{p})$. To achieve the desired monotone dependence of F on the spatial scale as well as on the distance as discussed in the previous paragraph, we make it proportional to a function $d_s(\mathbf{r_{pq}}, \sigma_s(\mathbf{p}))$, where $d_s(\mathbf{r_{pq}}, \sigma_s(\mathbf{p}))$ is a nonnegative, nonincreasing function of $\|\mathbf{r}\|$, not identically 0 for $\|\mathbf{r_{pq}}\| \leq \sigma_s(\mathbf{p})$, and 0 for $\|\mathbf{r_{pq}}\| > \sigma_s(\mathbf{p})$.

Just as the spatial scale parameter is chosen to ensure that F at $\mathbf{p}$ depends on spatial structure in the neighborhood of $\mathbf{p}$, the photometric (contrast, or gray level) scale parameter at P is determined by the degree of gray level homogeneity and contrast of a specific region of interest containing P. It ensures that F at $\mathbf{p}$ is determined by points within the region, i.e., the point experiences negligible attraction to another point having gray level difference larger than a cut-off value characteristic of within-region gray level variability. To achieve this, we make F proportional to the function $d_g(\Delta I, \sigma_g(\mathbf{p}))$, where ΔI denotes the absolute gray level difference between points P and Q, and $\sigma_g(\mathbf{p})$ is the photometric scale parameter. $d_g(\Delta I, \sigma_g(\mathbf{p}))$ is a nonnegative, nonincreasing and symmetric function of ΔI, not identically 0 for $\Delta I \leq \sigma_g(\mathbf{p})$, and 0 for $\Delta I > \sigma_g(\mathbf{p})$.

Thus, $\mathbf{F_p}$ can be written as defined earlier in Equation (1), namely,

$$\mathbf{F_p} = \int_{\mathbf{q} \neq \mathbf{p}} d_s(\mathbf{r_{pq}}, \sigma_s(\mathbf{p})) d_g(\Delta I, \sigma_g(\mathbf{p})) \hat{\mathbf{r}}_{\mathbf{pq}} \mathbf{dq}$$

Two observations follow from the above definition of F. First, the scale parameters at different points within a region R are mutually dependent since they are all determined by the structure of R. In particular, σ_s at a point P inside a region R depends on P's location relative to R's boundary, and all points inside R have the same σ_g value which corresponds to the contrast of R with its surround. For example, in a piecewise constant image, σ_s varies continuously and σ_g is piecewise constant. Second, since the region structure is to be determined in the first place, σ_s and σ_g are *a priori* unknown to the segmentation algorithm and must be computed at each point. Since a point in general belongs to many different regions at different scales, σ_s and σ_g will have multiple values at a point.

Note that when a homogeneous region (or background) is at least partly enclosed by the image border, the computation of F will be undefined at those image points P whose nearest border point is on image boundary. This is because the computation of $\sigma_s(\mathbf{p})$ involves points at distances $> \delta(\mathbf{p})$ from $\mathbf{p}$ (Fig. 1) and some such points are outside the image. To resolve this problem, we will treat the entire image as surrounded by a hypothetical, constant-value region whose contrast relative to the given image is infinite. Accordingly, for computational purposes, the points outside the image will be assumed to be accessible but having a gray level of infinity. This would yield the given, finite size image as the largest and least homogeneous region within the hierarchical segmentation of the hypothetical, infinitely large image.

3.3 Properties

In this section, we present some properties of $\mathbf{F}$ which collectively describe the relationship between the spatial structure of $\mathbf{F}$ and the image structure, and consequently, suggest $\mathbf{F}$ as a means of image segmentation. For brevity, the proofs are outlined for only some of the properties; the rest can be found in [2].

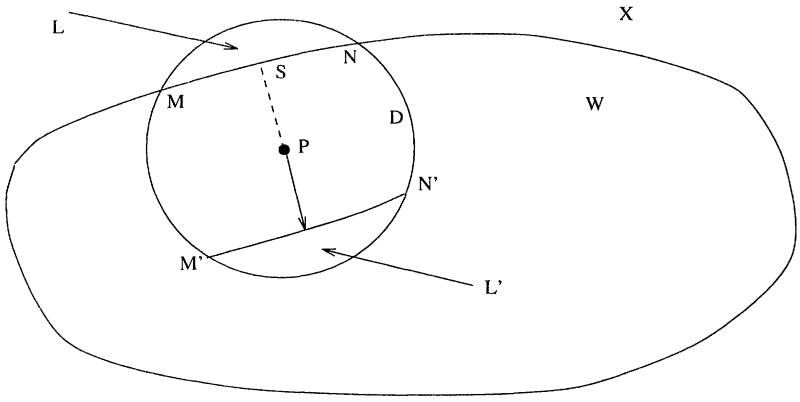


Figure 2. The force at a point P inside a homogeneous region points inward (Property 2).

1. Null Response: Suppose an image contains a constant-value disk, an arbitrary number of arbitrarily located other regions not intersecting with the disk, and a constant-value background. Then the value of $\mathbf{F}$ is zero at the disk center.

2. Inward Attraction: Let P be a point inside a homogeneous region W. Let D be the disk of radius $\sigma_s(\mathbf{p})$ centered at P which intersects with W and its border with a homogeneous region X of contrast C. Let M and N be the two points where the borders of D and W intersect (there will be exactly two such points by definition of σ_s). Then, for $\sigma_g < C$, there exists a point S along W's border within D and between the points M and N such that the direction of $\mathbf{F_p}$ at P is given by the vector from S to P, i.e., $\mathbf{F_p} = k\hat{\mathbf{r}}_{\mathbf{sp}}$, where k is a positive constant and $\hat{\mathbf{r}}_{\mathbf{sp}}$ denotes the unit vector from S to P. Thus $\mathbf{F_p}$ points inward.

Proof: Let L denote the region of intersection between D and X (Fig. 2). If we construct another region L' such that L and L' are symmetric about P, with the segment M'N' being symmetric to MN, then the force at P due to the neighborhood (D-L-L') will be 0 by symmetry (see Property 1). Therefore, the force $\mathbf{F_p}$ at $\mathbf{p}$ is due to the points in the regions L$\cup$L' and is given by,

$$\mathbf{F_p} = \mathbf{F}_{\mathbf{p}L} + \mathbf{F}_{\mathbf{p}L'} \tag{3}$$

where $\mathbf{F}_{\mathbf{p}L}$ and $\mathbf{F}_{\mathbf{p}L'}$ denote the forces at $\mathbf{p}$ due to regions L and L', respectively. Now for all $\mathbf{q}_1 \in L$, it is given that $\Delta I(\mathbf{p}, \mathbf{q}_1) > \sigma_g$. Therefore, $d_g(\Delta I(\mathbf{p}, \mathbf{q}_1), \sigma_g) = 0$ from the definition of d_g. For all $\mathbf{q}_2 \in L'$, $\Delta I(\mathbf{p}, \mathbf{q}_2) = 0 < \sigma_g$. Since d_g is a nonincreasing function of ΔI not identically 0, and $\Delta I(\mathbf{p}, \mathbf{q}_2 \in L') = 0 < \sigma_g < \Delta I(\mathbf{p}, \mathbf{q}_1 \in L)$, we have

$$d_g(\Delta I(\mathbf{p}, \mathbf{q}_2), \sigma_g) > 0 = d_g(\Delta I(\mathbf{p}, \mathbf{q}_1), \sigma_g). \tag{4}$$

Thus $\mathbf{F}_{\mathbf{p}L} = 0$, and consequently, $\mathbf{F_p} = \mathbf{F}_{\mathbf{p}L'}$. Since d_s is not identically 0 and nonnegative, it follows from Equations (1) (3) and (4) that $\mathbf{F}_{\mathbf{p}L'} > 0$. Therefore, $\mathbf{F_p} = \mathbf{F}_{\mathbf{p}L'} > 0$. Since $\mathbf{F}_{\mathbf{p}L'}$ is the net force on P due to all points $\mathbf{q_2} \in$ L', the direction of $\mathbf{F}_{\mathbf{p}L'}$ must be given by the unit vector $\hat{\mathbf{r}}_{\mathbf{ps}''}$ for some point S" $\in$ L'. If S' denotes a point of intersection of the line joining P and S" with the segment M'N', then the direction of $\mathbf{F_p}$ is given by

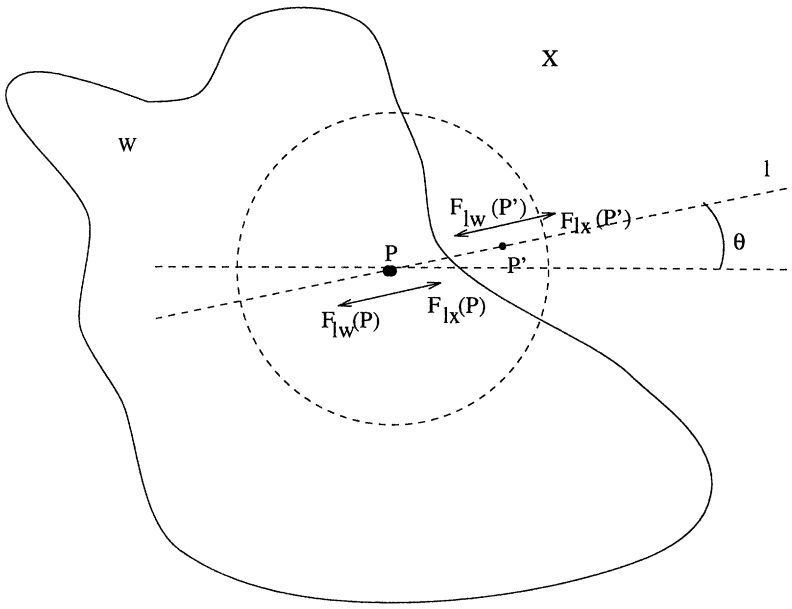


Figure 3. Divergence. If a point P in region W and another point P' in region X are infinitesimally far from each other, and hence also from the boundary between W and X, then the force vectors at P and P' are equal and opposite (Property 3).

the unit vector $\hat{\mathbf{r}}_{\mathbf{ps'}}$. Therefore, $\mathbf{F_p} = k\hat{\mathbf{r}}_{\mathbf{ps'}}$, where k is a constant. But for each point $\mathbf{s'}$ on segment M'N', there is another point $\mathbf{s}$ on segment MN where the line PS intersects the segment MN, and $\hat{\mathbf{r}}_{\mathbf{ps'}} = \hat{\mathbf{r}}_{\mathbf{sp}}$. Therefore, $\mathbf{F_p} = k\hat{\mathbf{r}}_{\mathbf{sp}}$.

3. Divergence: Consider a pair of points P and P' inside and outside, respectively, a homogeneous region W, and infinitesimally close to W's boundary. Then $\mathbf{F_p X F_{p'}} = \mathbf{0}$, i.e., the force vector undergoes direction reversal across the boundary, regardless of the shape of the boundary.

Proof: Let P' belong to region X (Fig. 3). Consider the line l through P and P', and the forces at P and P' due to points along l and within a distance $\sigma_s(\mathbf{p})(= \sigma_s(\mathbf{p}'))$. Let $F_{lW}(P)$ denote the total attraction force at P along l due to and towards the points inside W. Let $F_{lX}(P)$ denote the total attraction force at P along l due to and towards the points inside X. Analogously, let F_{lX} (P') and F_{lW} (P') denote the total attraction forces on P' along l due to and towards the points inside X and W, respectively. Then the total attraction force $\mathbf{F}_{lW}$ (P) at P towards W is given by $\mathbf{F}_{lW}(P) = (F_{lW}(P) - F_{lX}(P))\hat{\mathbf{w}}$ where $\hat{\mathbf{w}}$ is the unit vector along l and towards W. Similarly, the total attraction force $\mathbf{F}_{lX}$ (P') at P' towards X is given by $\mathbf{F}_{lX}(P') = (F_{lX}(P') - F_{lW}(P'))\hat{\mathbf{x}}$ where $\hat{\mathbf{x}}$ is the unit vector along l and towards X. Since each of W and X is a constant-gray-level region, we have $F_{lW}(P) = F_{lX}(P')$ and further since $\mathbf{F}$ is an even function of gray level difference, $F_{lX}(P) = F_{lW}(P')$. Therefore, $\mathbf{F}_{lW}(P) = -\mathbf{F}_{lX}(P')$. Now the total force $\mathbf{F}_W(P)$ at P towards W is given by $\mathbf{F}_W(P) = \int_0^\pi \mathbf{F}_{lW}(P)d\theta$ and the total force $\mathbf{F}_X$ (P') at P' towards

X is given by $\mathbf{F}_X(P') = \int_0^\pi \mathbf{F}_{lX}(P')d\theta$. From the previous three equations we see that, $\mathbf{F}_W(P) = -\mathbf{F}_X(P')$, which means that there is a directional discontinuity of magnitude π between the force vectors at P and P'.

4. Orthogonality: Consider two points P and Q inside a homogeneous region W such that the unit vector $\hat{\mathbf{r}}_{\mathbf{pq}}$ is orthogonal to the region boundary at the point S where the line PQ intersects region boundary, and is directed into W. If the intercept of W's boundary with the disk D of radius $r = \sigma_s(\mathbf{p})$ centered at P is symmteric about the line PQ, then the direction of $\mathbf{F}_\mathbf{p}$ is given by $\hat{\mathbf{r}}_{\mathbf{pq}}$, i.e., $\mathbf{F}_\mathbf{p} = k\hat{\mathbf{r}}_{\mathbf{pq}}$ for some positive constant k, i.e., $\mathbf{F}_\mathbf{p}$ is orthogonal to the boundary.

5. Medial Axis: At all points P along the medial axis of a region, $\mathbf{F}_\mathbf{p} = \mathbf{0}$.

Proof: This follows directly from the definition of σ_s. For any point P along the medial axis, $\delta(\mathbf{p}) = \delta'(\mathbf{p})$. Therefore, $\sigma_s(\mathbf{p}) = \delta(\mathbf{p})$. Since the force at P due to points farther than $\sigma_s(\mathbf{p})$ is negligible (from the definition of $\sigma_s(\mathbf{p})$) and since all points within the distance $\sigma_s(\mathbf{p})$ form a homogeneous region, it follows from Property 1 that $\mathbf{F}_\mathbf{p} = 0$. (Note that here the detected axis is medial in the sense of [6]. Other related shape axes are based on local symmetery [8] or inertia [33]).

6. Convergence: At points near and on either side of the medial axis of a region R, $\mathbf{F}$ is directed towards the medial axis. That is, $\mathbf{F}$ points away from the closest border segment in the sense described in Property 2.

7. Smoothness: If $d_g(\Delta I, \sigma_g)$ is a continuous function of ΔI, and $d_s(\mathbf{r}, \sigma_s)$ is a continuous function of $\mathbf{r}$ and σ_s, then $\mathbf{F}$ is a spatially continuous function at all nonboundary points of a region.

Proof: Consider a point P inside a completely homogenous region R, and another point T inside R and arbitrarily close to P. Then,

$$\mathbf{F}_\mathbf{p} = \int_{\mathbf{q}\neq\mathbf{p}} d_s(\mathbf{r}_{\mathbf{pq}}, \sigma_s(\mathbf{p}))d_g(\Delta I(\mathbf{p},\mathbf{t}), \sigma_g(\mathbf{p}))\hat{\mathbf{r}}_{\mathbf{pq}}d\mathbf{q}$$

and

$$\mathbf{F}_\mathbf{t} = \int_{\mathbf{q}\neq\mathbf{t}} d_s(\mathbf{r}_{\mathbf{tq}}, \sigma_s(\mathbf{t}))d_g(\Delta I(\mathbf{t},\mathbf{q}), \sigma_g(\mathbf{t}))\hat{\mathbf{r}}_{\mathbf{tq}}d\mathbf{q}$$

Now for any third point Q, $\Delta I(\mathbf{p},\mathbf{q}) = \Delta I(\mathbf{t},\mathbf{q})$ since P and T have the same gray level. Further, since P and T are arbitrarily close, $(\mathbf{r}_{\mathbf{tq}} - \mathbf{r}_{\mathbf{pq}})$ is arbitrarily small. Also, $\sigma_s(\mathbf{t}) - \sigma_s(\mathbf{p})$ is arbitrarily small from the definition of σ_s. Therefore, for any given choice of $\sigma_g(\mathbf{p})(= \sigma_g(\mathbf{t}))$, $(\mathbf{F}_\mathbf{p} - \mathbf{F}_\mathbf{t})$ is arbitrarily small. Consequently, $\mathbf{F}_\mathbf{p}$ is continuous everywhere within a region.

8. Closure: For any piecewise constant image, the contours along which $\mathbf{F}$ exhibits divergence are closed.

3.4 A Region's Signatures

The above properties of the transform collectively suggest the following structure of $\mathbf{F}$ associated with an image region R, or the signatures of R in the $\mathbf{F}$-field. When a σ_g value

corresponding to R's contrast is used to compute $\mathbf{F}$, it leads to much attraction between pixels within R and little attraction between pixels across R's boundary. Further, if the σ_s value at each point is chosen corresponding to R, then R is characterized by a spatially smooth and inward force flow, where the force lines emanate from the region border and converge at R's medial axis given by $\mathbf{F} = 0$. The direction of $\mathbf{F}$ undergoes a divergent discontinuity of magnitude π across the entire region border except at those border points where $\mathbf{F} = 0$, i.e., border points which also lie on the medial axis such as corner points. Thus $\mathbf{F}$'s magnitude along a region's border varies but is 0 at corners.

A test of $\mathbf{F}$'s validity can be performed by considering the nature of $\mathbf{F_p}$ at a point P inside an arbitrary shaped region R as R shrinks in size uniformly and vanishes. Let us first consider the case where P belongs to the medial axis of R. Here, $\delta(\mathbf{p}) = \sigma_s(\mathbf{p}) = \delta'(\mathbf{p})$. Since $\mathbf{F_p}$ is computed over a constant-value disk of radius $\sigma_s(\mathbf{p})$, $\mathbf{F_p} = 0$. Now consider, a point P off the medial axis of R. Then, $\delta(\mathbf{p}) < \sigma_s(\mathbf{p}) < \delta'(\mathbf{p})$. However, as R continues to shrink, the area of the region L' (= area of region L) in Fig. 2 approaches 0. Since the net force at P is that due to L' (see the proof of Property 2),

$$\lim_{Area(R)\to 0} \mathbf{F_p} = 0. \tag{5}$$

Thus, the influence on the $\mathbf{F}$ field caused by an image region vanishes as the area of the region vanishes, as is to be expected.

3.5 Estimation of Scale Parameters

Recall that image regions are in general recursively embedded, each standing in contrast with its surround and characterized by its own gray level homogeneity. An image point is associated with multiple regions, and therefore with multiple degrees of homogeneity and contrast scales. Since in the proposed transform an increasing value of σ_g corresponds to increasingly nonuniform regions, σ_g comprises one index into the structural hierarchy. Further, since each region has its own shape and size, and the spatial scale parameter at a point depends on its location with respect to region boundary, each image point is also associated with a number of spatial scales (σ_s values). Therefore, each image point is associated with a number of (σ_s, σ_g) pairs, corresponding to the different regions that it belongs to. In real images, a point is contained in only a small number of regions, and is therefore characterized by a small number of scale pairs. $\mathbf{F}$ makes explicit at each image location regions corresponding to all scales present at that location. A particular selection of regions across the image corresponds to a specific cutset of the segmentation hierarchy.

The signatures of a region in the $\mathbf{F}$ field, as described in the previous subsection, are obtained assuming that $\mathbf{F}$ at each point is computed using the appropriate pair of (σ_s, σ_g) values. We will now explain how these unknown scales can be estimated to yield the multiscale image structure. For estimation of these values, we will treat σ_s and σ_g as variables and identify their values that correspond to image regions. Suppose that for all images of interest, $(\sigma_s)_{\min} \leq \sigma_s \leq (\sigma_s)_{\max}$ and $(\sigma_g)_{\min} \leq \sigma_g \leq (\sigma_g)_{\max}$. That is, the ranges of sizes and contrasts to be encountered in images have known bounds. If no specific information is available for images to be processed, the image size and the maximum gray level can be used as $(\sigma_s)_{\max}$ and $(\sigma_g)_{\max}$, respectively, while $(\sigma_s)_{\min} = (\sigma_g)_{\min} = 1$ can be used as the minimum size and contrast a region could have. Suppose the transform is used to compute the force at each point for $(\sigma_s)_{\min} \leq \sigma_s \leq (\sigma_s)_{\max}$, and $(\sigma_g)_{\min} \leq \sigma_g \leq (\sigma_g)_{\max}$. Suppose that at a point P, the pair of values $(\sigma_{s1}, \sigma_{g1})$ and $(\sigma_{s2}, \sigma_{g2})$ correspond to two regions R1 and R2 at two adjacent scales which contain P, with R2 containing (>) R1, and $\sigma_{g2} > \sigma_{g1}$. That is, there is no other region R3 such that R1 < R3 < R2. By our

definition of scale given in Sec. 1, R1 and R2 represent two adjacent natural scales occuring in the image at P. Let $\mathbf{V_1}$ and $\mathbf{V_2}$ denote, respectively, the values of $\mathbf{F}$ corresponding to $(\sigma_{s1}, \sigma_{g1})$ and $(\sigma_{s2}, \sigma_{g2})$. As the value of the variable σ_s is increased from σ_{s1}, the image area responsible for a nonzero value of $\mathbf{F}$ at P (area L in Fig. 2) will increase, resulting in an increase in the magnitude of $\mathbf{F}$. Now from Property 2, as the value of σ_s at a point P increases beyond $\sigma_s(\mathbf{p})$, the shape of L changes gradually and $\mathbf{p}$ moves along the subregion boundary, resulting in a gradual change in $\hat{\mathbf{r}}_{\mathbf{sp}}$ A similar gradual change in $\mathbf{F}$ will also be associated with an increase in σ_g. Thus, $\mathbf{F}$ will slowly deviate from $\mathbf{V_1}$ as (σ_s, σ_g) values increase. However, for a sufficiently large value of σ_s, i.e., $\sigma_s > \delta'(\mathbf{p})$, the value of $\mathbf{F}$ will begin to depend on multiple disconnected components of R1 which are subregions of R2. Since the direction $\hat{\mathbf{r}}_{\mathbf{sp}}$ is in general different for subregions R1 and R2, $\mathbf{V_1}$ and $\mathbf{V_2}$ are in general different. Similarly, for $\sigma_{g1} << \sigma_g << \sigma_{g2}$, the subregions of R2 will make significant contribution to $\mathbf{F}$ value at P. Consequently, $\mathbf{F}$ will change with (σ_s, σ_g). As (σ_s, σ_g) approach $(\sigma_{s2}, \sigma_{g2})$, $\mathbf{F}$ will assume the relatively stable value of $\mathbf{V_2}$. Therefore, in the $\sigma_s\sigma_g$-space, the locations where $\mathbf{F}$ is stable will be scattered, associated with structures at different pairs of scale values. Somewhere between each pair of nearby locations of stable points in the $\sigma_s\sigma_g$-space corresponding to R1 and R2, $\mathbf{F}$ will make a sharper transition from the value $\mathbf{V_1}$ to $\mathbf{V_2}$. By traversing the $\sigma_s\sigma_g$-space, computing $\mathbf{F}$ using all σ_s, σ_g values at all image points, and identifying those parts in the $\sigma_s\sigma_g$-space where $\mathbf{F}$ is locally stable (has locally minimal variation), we can determine all the scales associated with P. The scale values at all image points can be estimated jointly, because together the values comprise the signatures of regions as explained in the previous subsection. Such scale estimation is robust for two reasons. First, at each point only qualitative changes in $\mathbf{F}$ are detected. Second, the qualitative changes at different points are analyzed jointly to detect the spatial signatures of a region. This further suppresses any noise in $\mathbf{F}$ which is already low because of the large neighborhoods used in the computation of $\mathbf{F}$.

The point pattern in $\sigma_s\sigma_g$-space defined by the locations of the actual scale values corresponding to any image point is unique for the point and the image, and represents the *a priori* unknown multiscale structure determined by the algorithm at the point. This structure is restricted for common images because an image point does not have multiple contrasts associated with the same spatial scale although it may be contained in multiple regions of the same contrast. Therefore, it would suffice to perform a linear sweep of the $\sigma_s\sigma_g$-space in the σ_g direction, while identifying all (if any) σ_s values which yield locally stable $\mathbf{F}$ for each σ_g.

3.6 Region Detection

For each region in the image, occuring at any scale, the automatic scale estimation process computes at each pixel in and around the region a $\sigma_s - \sigma_g$ pair of values which correspond to the region's shape (σ_s) and contrast (σ_g). When the $\mathbf{F}$ field is computed using these $\sigma_s - \sigma_g$ values, the field contains the signatures of the region. The detection of regions would thus require partitioning of the $\mathbf{F}$ field such that each cell has the signatures of a region.

A simple approach to finding candidate regions is to locate contours of force divergence. This is easy and the result robust since the directional discontinuity across such contours is known to be π. All characteristics comprising the region's signature may then be matched jointly with the local $\mathbf{F}$-field around the candidate regions to test the region hypothesis. The details of such hypothesis formation and testing are outside the scope of this paper.

3.7 Segmentation Hierarchy

The properties of the transform and the capability of automatic estimation of the scale parameters discussed in the previous subsections allow the construction of a hierarchical representation of segmentation. The computed scale values for nearby pixels are mutually compatible in that for all points within a region, σ_s values are continuous and σ_g values are constant.

The regions of a single gray level form the smallest regions. These may be defined as the leaf nodes of the hierarchy. Homogeneous regions having larger sizes or gray level variations are used to define higher levels. For example, the hierarchy may be based on spatial containment relationship; thus, the regions containing a particular leaf region are arranged in increasing order of size to define the path from the leaf node to the root. Nearby pixels merge into increasingly large regions as the path to the root is traversed. The subtree below any node in the hierarchy is unrelated to any other disjoint subtree, i.e., the structure, path lengths and σ_s-σ_g values associated with the nodes are unrelated across the subtrees. They reflect the *a priori* unknown spatial structure within an image. In this sense, the hierarchy define a recursive partition of the image into arbitrarily shaped regions, analogous to the irregular pyramid representation of [24]. Specific algorithms to compute the scale parameters and to obtain different hierarchical representations are not within the scope of this paper; these will be reported in subsequent publications.

4 Performance Analysis

Let us first review the overall performance of the transform with respect to the desired characteristics (A-D) listed in Section 2.2. Characteristics A, invariance to local edge geometry and topology, serves as a key motivation for proposing the transform, and is central to its design. The discussion in Sec. 3 makes it clear how this characteristic is possessed by **F**. For example, the capability of multiscale segmentation holds even if more than two regions share a border point since the properties of the transform leading to region signatures are not affected by shape and adjacency characteristics of the regions. With regard to desired characteristic B, the scale parameter σ_g provides a mechanism to accomplish contrast scaling. As σ_g increases, adjacent regions may merge. This is because the attraction of a point in one region from another point across region boundary may increase sufficiently so that the directional discontinuity in **F** responsible for the edge may vanish. Thus changing σ_g achieves the same result as contrast based split-and-merge of regions [15, 19], and therefore, the desired contrast scaling. Analogously, for any given σ_g, scale parameter σ_s helps achieve geometric scaling (desired characteristic C). Larger σ_s values at a point correspond to more global structures having a given contrast σ_g, which results in the capability to detect different spatial scales. The desired characteristic D is of course met since scales can be automatically estimated as explained in Sec. 3.5.

In the rest of this section, we will divide the performance of the transform into two types. The first type is concerned with the capabilities to detect off-axis signatures of a region, specifically those represented by Properties 2, 3, 7 and 8. The second type consists of region signatures related to the medial axis, represented by Properties 5 and 6. Properties 1 and 4 do not directly contribute to the region signatures. Sections 4.1 and 4.2 examine the type 1 and type 2 performance, respectively. In discussing each type of performance, we consider two types of deviations from the model of piecewise constancy that was often used in deriving the transform's properties. These deviations better characterize real images as stated earlier. We investigate the effect these deviations have on the particular aspects

of the region signatures, i.e. on the different properties. We then examine the impact of different choices of the function F, i.e., the functions d_s and d_g, on the transform's performance. No detailed proofs are given for these claims as doing so will require the use of specific models of the deviations and further analysis which are beyond the scope of this paper. Results of further, experimental evaluation of the performance can be found in [2].

4.1 Off-Axis Signatures

In this section, we examine the first type of performance of the transform, namely, the impact on inward attraction, divergence, smoothness and closure properties.

4.1.1 Deviation from Piecewise Constancy

In the discussion so far, we have assumed that the regions have a constant gray level. We will now explain how the segmentation performance of the transform extends to regions having other types of smooth variations. If the region does not have a constant intensity then the force at a point P in the region will include additional components due to differential rates of change of intensities in different directions away from P (unlike the case for Property 1). For the simple case of an intensity ramp, the changes in intensity around P are antisymmetric. Since force depends on the absolute intensity difference, P still experiences equal and opposite forces from radially symmteric locations within the region resulting in zero net force from these locations.

Regions in real images often contain shading which is more complex spatial variation of intensity than represented by the ramp considered above, e.g., given by a polynomial in image coordinates x and y. Consider a point P within such a region R and another point Q in R within a neighborhood of radius $\sigma_s(\mathbf{p})$ centered at P. Then, $d_g(\Delta I(\mathbf{p},\mathbf{q}))$ will vary for different points Q, unlike was the case for constant-value regions. This variation will in general be nonlinear, partly due to the nonlinear variation in $\Delta I(\mathbf{p},\mathbf{q})$. Now consider another point T within the neighborhood but not within R. If the range of $\Delta I(\mathbf{p},\mathbf{q})$ values is sufficiently small compared to $\Delta I(\mathbf{p},\mathbf{t})$ for all choices of Q and T, then many of the properties of the transform may still hold. The proofs of Properties 2, 3, 7 and 8 given earlier suggest that the boundaries of the regions may still be detected as before. Accordingly, at a region boundary, there will still be directional discontinuities because of the large gray level discontinuity; and in the process of finding the scale parameter values for the region, stable response of $\mathbf{F}$ will be found for the same values of σ_s and σ_g as if the region were homogeneous. Any directional discontinuities found at locations other than the region border, due to nonlinear variation in gray level, will not persist if σ_g is varied. Since the gray levels vary smoothly within the region, Property 7 suggests that F will still be continuous within the region although it will exhibit differences in F values from the piecewise constant case. Therefore, regions with shading but in contrast with the surround should still have signatures similar to those for the piecewise constant case. Verification of the above extrapolation of the properties and the exact restatement of these and other properties for the general case would require exact models of within-region intensity variation, and will be omitted here.

4.1.2 Intensity Noise

We will now consider sensitivity to noise in intensity values. First, suppose that the regions have constant values but contain independently distributed, zero-mean, additive noise having a distribution which is symmetric with respect to the mean. Consider a point P and any

other point Q within a neighborhood of radius $\sigma_s(\mathbf{p})$ around P. Then, $d_g(\Delta I(\mathbf{p},\mathbf{q}))$ will have the same mean as for the case when the noise is absent. Since $d_g(\Delta I(\mathbf{p},\mathbf{q}))$ is a symmetric function of $\Delta I(\mathbf{p},\mathbf{q}))$, the expected value of $\mathbf{F}(\mathbf{p},\mathbf{q})$ will remain unchanged compared to the case without noise. Therefore, the expected value of $\mathbf{F_p}$ due to all points $\mathbf{q}$ in the image is the same with or without noise. Now suppose that the regions exhibit ramp-like intensity variation which is contaminated by independent, zero-mean, additive noise. Again, because of gray level antisymmetry about P, and the symmetry of $d_g(\Delta I)$ with respect to ΔI, the region boundary will remain unchanged assuming the region contrast with the surround is high compared to within region variation.

For shaded regions, the noise effects will be anisotropic because the region intensities are asymmetric. For a given point P, consider two other points Q and T within a neighborhood of radius $\sigma_s(\mathbf{p})$ around P. Q is within the region R but T is across R's border. If the range of $\Delta I(\mathbf{p},\mathbf{q})$ values is sufficiently small compared to the range of $\Delta I(\mathbf{p},\mathbf{t})$ values, then resulting $\mathbf{F}$ will have limited differences relative to the noiseless case. That is, boundaries of regions with shading but in contrast with the surround will still coincide with direction discontinuities in $\mathbf{F}$ and will therefore still be detected. However, as for noiseless shaded regions, exact analysis is necessary to obtain the true characterization of the region signature and its dependence on noise which we will again omit in this paper.

4.1.3 Choices of F

While defining the transform (Sec. 3), we stated that $d_s(\mathbf{r_{pq}}, \sigma_s(\mathbf{p}))$ should be a nonincreasing function of the magnitude of $\mathbf{r_{pq}}$, and $d_g(\Delta I, \sigma_g(\mathbf{p}))$ should be a nonincreasing and symmetric function of ΔI. A variety of such functions could be used including pulse (box-car), Gaussian, exponential, and linear functions. For example, we may use a Gaussian for d_s as well as d_g, having standard deviations of $k_s\sigma_\mathbf{s}$ and $k_g\sigma_\mathbf{g}$, resepectively, where k_s and k_g are normalization constants. The choice of Gaussian for d_s and d_g results in optimal localization properties in both spatial and transform domains, in addition to others such as separability in computation. Then, the transform at image location $\mathbf{p}$ is given by

$$\mathbf{F_p} = \int_{\mathbf{q}\neq\mathbf{p}} e^{-\frac{\|\mathbf{r_{pq}}\|^2}{2(\mathbf{k_s}\sigma_\mathbf{s}(\mathbf{p}))^2}} e^{-\frac{\Delta I^2(\mathbf{p},\mathbf{q})}{2(\mathbf{k_g}\sigma_\mathbf{g}(\mathbf{p}))^2}} \hat{\mathbf{r}}_{\mathbf{pq}} \mathbf{dq} \tag{6}$$

The properties of the transform given in Sec. 3 hold for any choices of such functions. Although the exact values of the force vectors and the computational speeds depend on specific choices of the functions, the region signatures and segmentation hierarchy remain unchanged. We have verified empirically that this in fact is the case for the four choices of box-car, Gaussian, exponential and linear functions [2].

4.2 Axial Signatures

This section discusses the impact of the two kinds of deviations from piecewise constancy and the choices of F on the medial axis related signatures. There are two ways in which the transform yields multiscale description of region shape. First, of course, is through the detection of region boundaries which may be used to estimate the medial axis following its definition, or using existing algorithms [6, 31]. The reliability of the region axis detected by this method directly depends on the corresponding reliability of detected region boundaries which we have discussed above. The second, more direct way in which the transform extracts multiscale region shape information is by making explicit the location of the medial axis in the field signatures of the region. We will now consider the performance of this

method of medial axis detection with respect to shading and noise. As stated in Property 5, for constant-value regions the $\mathbf{F} = \mathbf{0}$ curve within the region where $\mathbf{F}$ directions converge represents the medial axis. Now in the vicinity of locations where $\mathbf{F} = \mathbf{0}$, the magnitude of $\mathbf{F}$ will change at a rate determined by how fast the neighborhood of radius σ_s begins to have significant intersection with an adjacent region, and thus on the the local region shape. The steeper the $\mathbf{F}$ variation, the more accurate will be the detection of the $\mathbf{F} = \mathbf{0}$ locations, and therefore, the medial axis. However, if the intensity value within the region is not constant, then the disk of radius $\sigma_s(\mathbf{p})$ centered at a point on the medial axis will not in general have isotropic intensity distribution about the point. The force at the point will not be 0, and the points where $\mathbf{F} = \mathbf{0}$ may be off the medial axis. Therefore, when shading is present, the detection of medial axis is less reliable than the detection of region borders. With regard to noise, there will be no expected change in results for noisy piecewise constant regions for reasons analogous to those given for off-axis signatures. However, in shaded regions, noise will further increase the medial axis deviation beyond that already present due to shading alone. This is because the function d_g in the integral defining the transform is in general nonlinear, and therefore, in the presence of shading, uniform noise distribution around the point will cause unequal deviations in the $\mathbf{F}$ value in different directions, which will lead to deviation in the location of the $\mathbf{F} = \mathbf{0}$ curve. Finally, let us consider the effect of different choices of F. For the piecewise constant case, changes in F will not cause any deviation in the mdeial axis because of symmetry. However, when shading is present different choices of the nonlinear function d_g will in general result in different distributions of force magnitudes and hence deviation in the $\mathbf{F} = \mathbf{0}$ curve.

5 Summary

We have introduced a transform for integrated detection of image edges and regions at all natural scales, and thus for general purpose low-level image segmentation. Our objective here is to present the transform and its capabilities; details and different applications are left to future work. The transform computes a force vector at each image point, such that the spatial distribution of vectors makes explicit region edges and medial axes at all scales. The points within and on the boundary of a region are attracted towards its interior, thus causing a directional discontinuity across the boundary. The medial axes also exhibit similar discontinuities in the force direction. An important property of the transform is that the transform space is the same as image space, unlike for example the Fourier transform. The resulting multiscale representation is directly useful for analysis by humans, e.g., image browsing, manipulation and retrieval. The use of scale parameters and the detection of structure from qualitative characteristics of -field makes the use of the transform robust and free of critical thresholds. We have presented the properties of the transform. Experimental results, not reported here, show that the segmentations provided by the transform have few errors even for images with large noise, intricate geometric structure, and shading. The transform could be used in a variety of applications where determination of perceptually salient image structure is critical or advantageous, such as image compression, motion analysis, texture analysis and perceptual grouping. We plan to report on some of these applications in the future. We have not given algorithms for using the transform to automatically estimate the scales and identify region signatures. These will also be reported later along with the applications. Since no assumptions specific to the sensing modality are made in defining the transform, it could be applied to other types of data such as range images and synthetics aperture radar images. It will be interesting to compare the transform with force based clustering algorithms [16] which we plan to do.

Acknowledgements

This support of the Advanced Research Projects Agency under grant N00014-93-1-1167 administered by the Office of Naval Research, and the National Science Foundation under grant IRI 93-19038 is gratefully acknowledged. Thanks are also due to the anonymous reviewers of another, journal version of this paper whose comments were very helpful.

References

[1] N. Ahuja. A transform for detection of multiscale image structure. In *Proceedings Computer Vision and Pattern Recognition*, pages 780–781, New York, 1993.

[2] N. Ahuja. A transform for multiscale image segmentation. Technical Report 96-01, Beckman Institute Technical Note, July 1995.

[3] N. Ahuja and J. Chuang. Shape representation using a generalized potential field model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1996. to appear.

[4] V. Berzins. Accuracy of lapacian edge detectors. *Computer Vision, Graphics and Image Processing*, pages 195–210, 1984.

[5] T. Binford. Inferring surfaces from images. *Artificial Intelligence*, pages 205–244.

[6] H. Blum. A transformation for extracting new descriptors of shape. In Walthen Dunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380. MIT Press, Cambridge, MA, 1967.

[7] R. Boomgaard and A. Smeulders. Towards a morphological scale-space theory. In *Shape in Picture: Mathematical Descriptions of Shape in Grey-Level Images*, pages 631–640. Springer-Verlag, New York, 1992.

[8] M. Brady and H. Asada. Smoothed local symmetries and their implementation. *International Journal of Robotics Research*, 3(3):36–61, 1984.

[9] J.F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 679–698, 1986.

[10] J. J. Clark. Authenticating edges produced by zero-crossing algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 43–57, January 1989.

[11] L.S. Davis. A survey of edge detection techniques. *Computer Graphics and Image Processing*, pages 248–270, 1975.

[12] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions Pattern Analysis and Machine Intelligence*, PAMI-6(11):721–741, 1984.

[13] R. M. Haralick. Digital step edges from zerocrossings of second directional derivative. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 58–68, 1984.

[14] R. M. Haralick and L. Shapiro. A survey of image segmentation algorithms. *Computer Vision, Graphics and Image Processing*, 1492.

[15] S.L. Horowitz and T. Pavlidis. Picture segmentation by a directed split and merge procedure. In *Proc. Second Intl. Joint Conf. on Pattern Recognition*, pages 424–433, 1974.

[16] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice-Hall, 1983.

[17] J. A. Koenderink. *Solid Shape.* MIT Press, 1990.

[18] T. Lindeberg. *Scale-Space Theory in Computer Vision.* Kluwer Academic, Boston, 1994.

[19] T. Lindeberg and J. O. Eklundh. Scale-space primal sketch: Construction and experiments. In *Image and Vision Computing*, pages 3–18. January 1992.

[20] Y. Lu and R.C. Jain. Behavior of edges in scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 337–356, 1989.

[21] J. Malik and P. Perona. Edge detection by diffusion. Technical report, University of California, Berkeley, 1986.

[22] D. Marr and E. Hildreth. A theory of edge detection. In *Royal Society of London*, volume B-207, 1980.

[23] F. Meyer and S. Beucher. Morphological segmentation. *Vis. Communication Image Representation*, 1(1):21–46, 1990.

[24] A. Montavert, P. Meer, and A. Rosenfeld. Hierarchical image analysis using irregular tessellations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 307–316, 1991.

[25] B. S. Morse, S. M. Pizer, and A. Liu. Multiscale medial analyis of medical images. In H. Barrett and A. Gmitro, editors, *Proceedings 14th International Conference on Information Processing in Medical Imaging.* Springer-Verlag, 1993.

[26] D. Mumford and J. Shah. Boundary detection by minimizing functionals, i. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 22–26, 1985.

[27] V. Nalwa and T. Binford. On detecting edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 699–714, 1986.

[28] V. S. Nalwa. Experiments with a spatiotemporal correlator. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 712–716, 1992.

[29] P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks and roofs. In *Third Intl. Conf. Computer Vision*, pages 52–57, Osaka, Japan, Dec. 1990.

[30] J. Ponce and M. Brady. Toward a surface primal sketch. Technical Report MIT-AI-TR-824, MIT, 1985.

[31] A. Rosenfeld and A. Kak. *Digital Picture Processing*, volume 2. Academic Press, 1981.

[32] A. Rosenfeld and M. Thurston. Edge and curve detection for visual scene analysis. *IEEE Transactions on Computers*, pages 562–569, 1971.

[33] J. Brian Subirana-Vilanova. Curved inertia frames and the skeleton sketch: Finding salient frames of reference. In *Proceedings IEEE International Conference on Computer Vision*, pages 702–708, 1990.

[34] R. Whitaker and S. Pizer. Geometry-based image segmentation using anisotropic diffusion. In *Shape in Picture: Mathematical Descriptions of Shape in Grey-Level Images*, pages 641–650. Springer-Verlag, New York, 1992.

[35] A. Witkin. Scale space filtering. In *International Joint Conference on Artificial Intelligence*, 1983.

[36] A. Yuille and T. Poggio. Scaling theorems for zero crossings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 15–25, 1986.

Image Segmentation Using Clustering

Anil K. Jain
Department of Computer Science
Michigan State University
East Lansing, MI 48824
jain@cps.msu.edu

Patrick J. Flynn
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA 99164
flynn@eecs.wsu.edu

Abstract

Pattern Recognition methodology has influenced many areas of computer vision research. This chapter addresses one of the most direct and long-lived applications of classification to image analysis, namely the use of unsupervised classification (cluster analysis) for image segmentation. The use of clustering for image segmentation dates back to the late 1960s and many of the techniques developed then are still in popular use today. This chapter outlines the segmentation and clustering problems and their close relationship, and surveys the use of clustering for segmentation in a variety of different domains (*e.g.*, intensity images, textured images, multispectral images, and range images).

1: Introduction

An *image segmentation* is a fundamental component in many high-level computer vision applications, and is typically viewed as an unsupervised classification task [1]. The segmentation of the image(s) presented to an image analysis system is critically dependent on the scene to be sensed, the imaging geometry, configuration, and sensor used to transduce the scene into a digital image, and ultimately the desired output (goal) of the system. Image segmentation procedures are often labeled as *edge-based* or *region-based*, referring in each case to a dominant computational philosophy of identifying homogeneous regions or contours of local inhomogeneity, respectively. An important region-based technique is *split–and–merge* segmentation [10]. Some attempts [18, 19] have also been made to merge the results of edge-based and region-based methods to improve segmentation results.

Clustering is a generic label for a variety of procedures designed to find "natural" groupings in *unlabeled* data presented as a set of *patterns*, each a list of *features*. Clustering is often referred to as *unsupervised classification* to reinforce the notion that the goal is to discover

0-8186-7644-2 $5.00 © 1996 IEEE

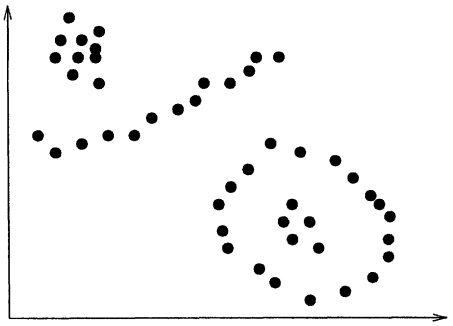


Figure 1. Cluster structures. Clusters can be compact or elongated, well-separated or poorly separated from other clusters.

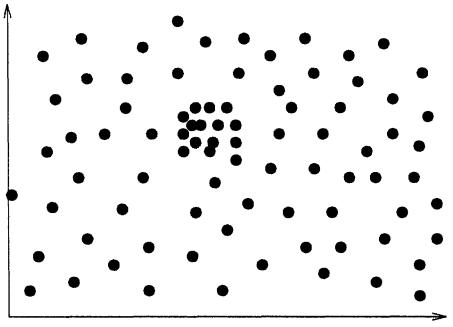


Figure 2. A "dot" cluster.

structure (classes or categories) in a multivariate data set without an explicit model for each pattern class, an oracle to supply correct or true classifications for representative samples, or an external procedure to validate the groupings produced by the clustering algorithm. Clustering is a difficult problem because "natural groupings" in multidimensional data can have very different properties, as illustrated in Figure 1. Another example of clustering (addressed by Sher and Rosenfeld [3]) is the detection and delineation of regions containing a high density of patterns compared to their background; Figure 2 shows such an image.

The applicability of clustering methodology to the image segmentation problem was recognized over three decades ago, and the paradigms underlying the initial pioneering efforts are still in use today. A recurring theme is to define feature vectors at every image location (pixel) composed of both functions of image intensity and functions of the pixel location itself. This basic idea has been successfully used for intensity images (with or without texture), range (depth) images and multispectral images. Figure 3 depicts the scheme. Contrary to many applications of pattern recognition, the output of a clustering-based segmenter is easily visualized as a labeling of the input image with (for example) different colors or gray scale shades representing different label values.

The goals of this chapter are as follows:

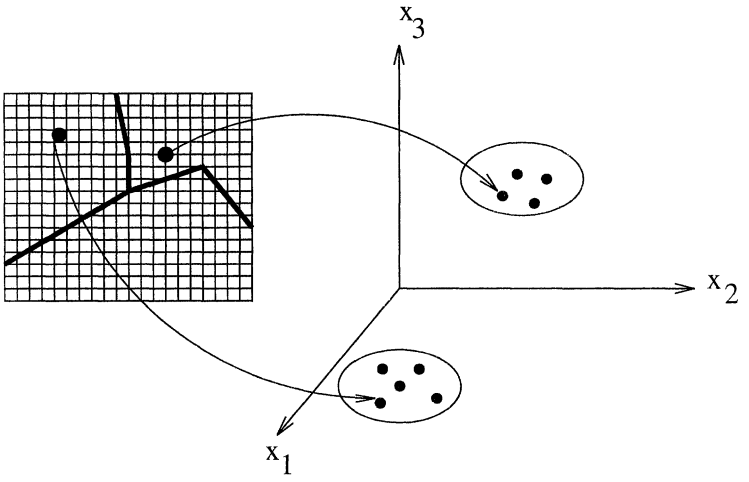


Figure 3. Feature representation for clustering. Image measurements and positions are transformed to features. Clusters in feature space correspond to image segments.

1. review image segmentation and the major paradigms used to obtain segmentations,
2. survey clustering methodology including its placement in the taxonomy of pattern recognition techniques,
3. motivate the application of clustering to segmentation in general,
4. recognize the work of those who pioneered this technique, and
5. describe selected applications of clustering to specific image segmentation domains over the last thirty years.

We make no effort to be exhaustive in this survey, but strive to mention those approaches that laid the foundation for the work of others or achieved significant milestones in the application of cluster analysis to image segmentation.

2: Segmentation: Definition, Intuition, and Major Paradigms

An *image segmentation* is typically defined as an exhaustive partitioning of an input image into regions, each of which is considered to be homogeneous with respect to some image property of interest (*e.g.*, intensity, color, or texture) [11]. If

$$\mathcal{I} = \{x_{ij}, i = 1 \ldots N_r, j = 1 \ldots N_c\}$$

is the input image with N_r rows and N_c columns and measurement value x_{ij} at pixel (i, j), then the segmentation can be expressed as $\mathcal{S} = \{S_1, \ldots S_k\}$, with the lth segment

$$S_l = \{(i_{l_1}, j_{l_1}), \ldots (i_{l_{N_l}}, j_{l_{N_l}})\}$$

consisting of a connected subset of the pixel coordinates. No two segments share any pixel locations ($S_i \cap S_j = \emptyset \quad \forall i \neq j$), and the union of all segments covers the entire image ($\cup_{i=1}^{k} S_i = \{1 \ldots N_r\} \times \{1 \ldots N_c\}$).

Jain and Dubes [12], after Fu and Mui [21], identify three techniques for producing segmentations from input imagery: *region-based*, *edge-based*, or *cluster-based*. Intuitively, we can describe region-based approaches as those which identify maximal homogeneous regions through the computation of image properties defined over candidate regions, edge-based approaches as those which employ local properties to locate points of discontinuity belonging to the edges between regions, and clustering-based approaches as those which identify compact and well-separated regions in a (typically multidimensional) feature space and segment the images based on both these regions and on image connectivity. The boundary between clustering-based and region-based techniques is not always clear since "compact and well-separated clusters" in the feature space could be part of a criterion for homogeneity employed in a region-based approach. However, since the focus of this chapter is cluster analysis and its applications we would consider any segmentation algorithm which employs a typical clustering algorithm as a major component in its pixel labeling scheme as clustering-based.

Consider the use of simple gray level thresholding to segment a high-contrast intensity image. Figure 4(a) shows a grayscale image of a textbook's bar code scanned on a flatbed scanner. Part (b) shows the results of a simple thresholding operation designed to separate the dark and light regions in the bar code area. Binarization steps like this are often performed in character recognition systems. Thresholding in effect 'clusters' the image pixels into groups based on the one-dimensional intensity measurement [2, 4]). A postprocessing step separates the classes into connected regions. While simple gray level thresholding is adequate in some carefully controlled image acquisition environments and much research has been devoted to appropriate methods for thresholding [22, 23], complex images require more elaborate segmentation techniques. For example, Ohlander [20] used multidimensional histograms for segmentation.

Segmentation, like many other vision tasks, is defined by its ultimate use, and the diversity of image types and noise sources has made image segmentation a fertile research area. This diversity has also established segmentation as a *difficult* research area since in many cases techniques learned in one application domain do not transfer with ease (or at all) to another domain. The evaluation of a segmented image is a topic that has not been addressed with regularity by researchers. To the extent that segmentation performance is subjectively measured, a methodological gap exists and should be addressed [23]. The nature and quality of the input imagery, as well as expectations on the characteristics of the output regions, dictate the set of valid segmentation strategies that should be attempted and perhaps even the details of their design and implementation.

A critical issue associated with segmentation is the image representation. This reduces to the problem of defining the feature set, its dimensionality, and its type and scale. In the segmentation techniques surveyed in this chapter, the image measurement can be intensity, range, or features derived from intensity or range (*e.g.*, texture energy at a specified scale and orientation, or an estimated normal to a surface), its dimensionality ranges from one to twenty-eight, its type can be discrete or continuous, and its scale can be ordinal or ratio (for a discussion of data type and scale in the context of clustering methodology, see Jain and Dubes [12]). Many segmenters use measurements which are both *spectral* (*e.g.*, the multispectral scanner used in remote sensing) and *spatial* (based on the pixel's location in the image plane). The measurement at each pixel will be viewed henceforth as a point in a multidimensional space (a *feature space*), where the nature of the space is described when needed. Figure 3 shows a three-dimensional feature space (x_1, x_2, x_3) corresponding to an image measurement.

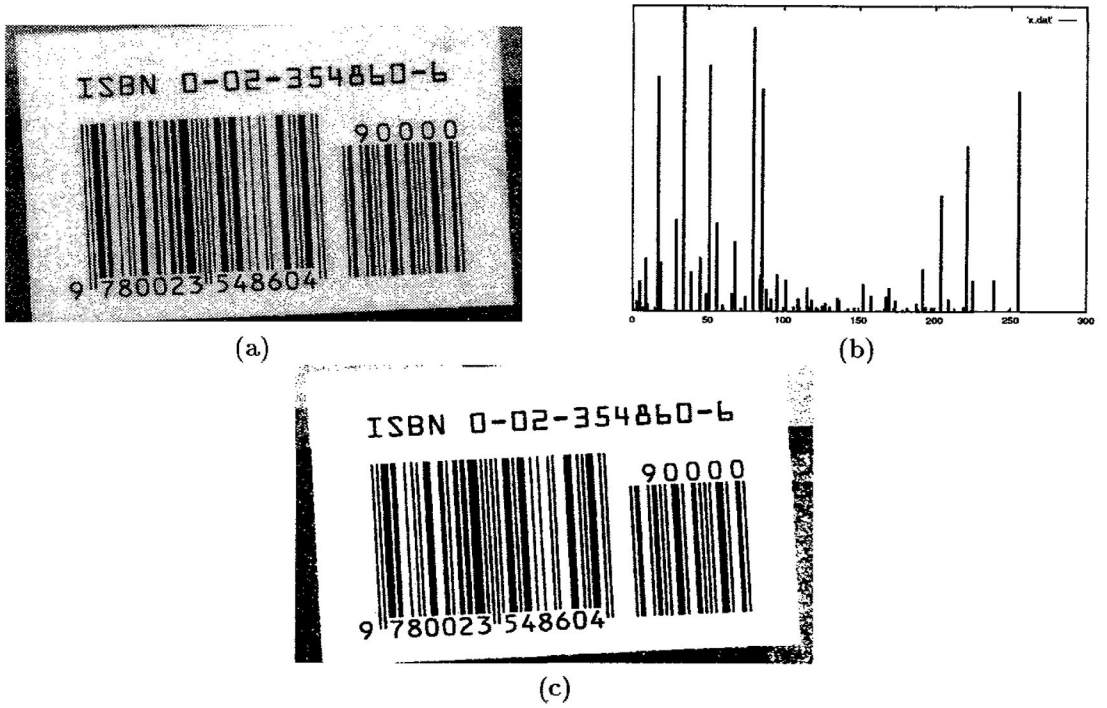


Figure 4. Binarization via thresholding. (a): Original grayscale image. (b): Gray-level histogram. (c): Results of thresholding.

3: Clustering Procedures: Context and Taxonomy

The essential elements of a clustering problem are a feature space $\mathbf{X}$ (typically multidimensional), samples (*patterns*) $X = \{\mathbf{x}_1, \ldots \mathbf{x}_n\}$ from that space, presumed cluster structure in the samples, and the absence of accessible class information for the samples. A clustering procedure is expected to (i) identify the number of clusters K present in the available samples, and (ii) obtain a label set $\mathbf{L} = \{l_1, \ldots l_n\}(l_i \in \{1 \ldots K\}, \forall i)$ for the samples. These two tasks may be performed simultaneously or sequentially; in some techniques K is specified to the algorithm by the user.

The combinatorics of the clustering problem make simple enumerative schemes impractical for all reasonably-sized problems. The number $S(n, K)$ of distinct clusterings of n objects into K clusters is a Stirling number of the second kind [12]:

$$S(n, K) = \frac{1}{K!} \sum_{i=1}^{K} (-1)^{K-i} \left(\begin{array}{c} K \\ i \end{array} \right) i^n$$

For example, while 34,105 distinct partitions of ten objects into four clusters exist, there are 11,259,666,000 partitions of nineteen objects into four clusters. If the number of clusters is unknown but can be bounded above by (say) K_{max}, the number of possible partitionings is $\sum_{K=1}^{K_{max}} S(n, K)$. Moreover, in order to select the best clustering a criterion would need to

be evaluated for each of the large numbers of candidates. Clearly, this approach is infeasible (along with variants which prune the space of reasonable candidate partitions), and in practice we sacrifice optimality of the partitions for computational efficiency.

The literature of clustering is vast and resides in many disciplines; likewise, the diversity of clustering techniques makes establishment of a comprehensive taxonomy of these techniques difficult. The accepted major dichotomy among clustering methods is the distinction between *partitional* and *hierarchical* approaches. In essence, a hierarchical clustering procedure produces a nested sequence of partitionings covering the spectrum between the trivial partition of n patterns into n singleton clusters and the trivial partition of n patterns into a single n-element cluster. Such hierarchical clusterings are often depicted graphically by a *dendrogram* which can be 'cut' at a desired level of proximity, yielding a clustering or a partition. In contrast, a partitional clustering is a single partition of the input pattern set into clusters. Examples of partitional clustering techniques are the well-known ISODATA [14] and CLUSTER [12] algorithms which use a squared-error criterion, and Zahn's graph-theoretic clustering procedure [49] which identifies disjoint clusters by breaking inconsistent edges in a minimum spanning tree of the pattern matrix.

The absence of labeled training data and the need for production of a set of labels for the input patterns establishes a close correspondence between the general clustering problem and the task of image segmentation. It is not surprising, then, that clustering has been repeatedly applied to image segmentation problems of many types, often with success. There is a rich body of literature on the segmentation problem, and clustering-based approaches have always been a popular choice for researchers. The keys to successful application of clustering to image segmentation are the specification of the feature space $\mathbf{X}$, establishment of an appropriate dissimilarity measure $d(\mathbf{x}_i, \mathbf{x}_j)$ for pairs of patterns in that space, and the choice of an appropriate algorithm for clustering this type of data.

The difficult issue of evaluation of segmentation results has received some attention over the years. The parallel issue in cluster analysis is *cluster validity*; while all clustering algorithms will produce a clustering of an input pattern set (and in general those clusterings will differ), how should the quality of a clustering be measured? Cluster validity has been extensively studied and offers some lessons to those interested in validating segmentations. A related issue that is not as germane to segmentation as validity is *cluster tendency*: does an input pattern set indeed contain cluster structure? The source of images in a segmentation application is typically contrived to avoid circumstances where only one segment (or homogeneous region) is present, and image structure (*e.g.*, image plane connectivity) makes the assignment of each pattern to its own cluster unreasonable.

Not all clustering problems contain compact and well-separated clusters. Yet hyperellipsoidal clusters are assumed to be present by many clustering techniques. A variety of methods for "cleaning" the input pattern set have been proposed over the years. Jolion and Rosenfeld [6] proposed to associate with each input pattern a weight that depends on the empirical density observed in that pattern's vicinity. Cluster statistics computed during clustering use these empirical weights and were demonstrated to reduce the bias caused by the inappropriate linkage of 'background noise' patterns into clusters.

Although clustering methodology is a mature field with many well-understood and powerful techniques to offer prospective users, research into new clustering techniques continues. For example, Mao and Jain [48] recently developed a self-organizing network for data clustering, and applied it to both traditional data sets from the statistical literature, and the examples from the texture segmentation problem (described below). The expectation maximization

(EM) technique [17, 55] for resolving mixture models yields a clustering algorithm which can be applied to image segmentation.

3.1: Squared Error Clustering

Squared-error clustering procedures are partitional clustering methods which attempt to minimize a squared-error objective function over the space of possible partitioning of the pattern set. The squared error for a clustering of n patterns into K clusters is

$$E_K^2 = \sum_{j=1}^{K} \sum_{i=1}^{n_j} \|\mathbf{x}_i^{(j)} - \mathbf{m}_j\|^2,$$

where $\mathbf{x}_i^{(j)}$ is the i^{th} pattern belonging to cluster j and

$$\mathbf{m}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} \mathbf{x}_i^{(j)}$$

is the mean vector of cluster j (formed from the n_j patterns assigned to it). The goal of squared error clustering algorithms is to find the partition which (for a given K) minimizes E_K^2. Intuitively, we can minimize E_K^2 (which is sometimes called the *within-cluster variation*) by obtaining compact and well-separated clusters.

Many squared-error algorithms proceed iteratively, beginning with an initial clustering chosen randomly or 'sensibly', with a goal of decreasing the squared error at each iteration. The initial partition affects the final clustering, and simple minimization procedures can become trapped in local minima of the squared-error objective function; a standard technique for overcoming this problem is to run the clustering program several times with different initial conditions on each run, and save that clustering which yields the minimum value of E_K^2. Additional heuristics allow the ability to adjust the number of clusters during the algorithm's operation; these adjustments typically split clusters with large variance and merge clusters which have close mean vectors. The behavior of the splitting and merging heuristics is controlled by user-settable thresholds.

Even with special treatment to avoid local minima, convergence of a squared-error partitional clustering algorithm does not necessarily yield a configuration with the globally minimal E_K^2. Convergence criteria used in many clustering programs include a maximum number of relabeling iterations, a lack of significant decreases in E_K^2, and a minimal number of label changes during an iteration.

We now present an outline of the CLUSTER algorithm, based on the description of Jain and Dubes [12]. This algorithm has been used successfully in many image segmentation problems. The objective of CLUSTER is to minimize E_K^2. In its most popular version it has a single user-specified parameter: K_{max}, the maximum number of clusters to consider. CLUSTER will produce a sequence of clusterings containing $2, 3, \ldots K_{max}$ clusters. During an initialization phase, a candidate set of $K_{max} - 1$ clusterings is created as follows. Cluster centers for the 2-cluster solution are chosen to be the centroid of the pattern set and the pattern farthest from that centroid. The clustering itself is obtained by minimum distance classification with respect to these two centers. The pattern farthest from its cluster center is chosen as the third cluster center. This process repeats until there are K_{max} distinct cluster centers.

The main portion of CLUSTER contains two phases which are repeated in sequence until a pass through both phases does not decrease E_K^2. The first phase is a classical K-means pass which adjusts cluster memberships in each clustering to decrease E_K^2; this K-means pass is repeated until no class labels change or a maximum number of iterations is performed. The second phase of the algorithm is a forcing pass which merges clusters in pairs to see if a better clustering can be achieved. After the K-means and forcing passes complete, the squared error of the clustering is compared with its previous value, and the process repeats if the squared error decreased. When CLUSTER completes, a number of summary statistics are produced to aid in the (difficult) task of choosing the appropriate number of clusters.

3.2: Clustering Applications in Image Analysis

While the focus of this chapter is on the use of clustering techniques for the task of image segmentation, clustering has seen widespread use for other problems in image analysis. Perceptual grouping tasks (usually motivated by Gestalt principles and often viewed as an *intermediate-level* visual module) often make use of either traditional or *ad hoc* clustering procedures. Rosenfeld and Lee [8] proposed a heuristic for deriving connectivity graphs from line drawings. Scher *et al.* [7] developed a clustering procedure to group small collinear line segments into larger segments. The analysis of dot patterns (and the identification and representation of clusters of dots in digital imagery) was addressed by Velasco and Rosenfeld [9] and Ahuja and Tuceryan [50]; this latter work included an explicit comparison of specialized rules with classical clustering algorithms applied to the same dot patterns.

4: Image Segmentation Via Clustering

How has clustering been applied to image segmentation? In the remainder of this chapter, we will survey, compare, and contrast the variety of segmentation contexts in which clustering has been applied.

4.1: Intensity Image Segmentation

Schachter, Davis, and Rosenfeld [5] applied a local feature clustering approach to segmentation of gray-scale images. This paper emphasized the appropriate selection of features at each pixel rather than the clustering methodology, and proposed the use of image plane coordinates (spatial information) as additional features to be employed in clustering-based segmentation. For example, this paper succinctly shows that intensity should not be supplemented by simple edge features when constructing a feature space for clustering-based segmentation, since the edge features display little global variation. The goal of clustering was to obtain a sequence of hyperellipsoidal clusters starting with cluster centers positioned at maximum density locations in the pattern space, and growing clusters about these centers until a χ^2 test for goodness of fit was violated. A variety of features were discussed and applied to both grayscale and color imagery.

Silverman and Cooper [27] employed an agglomerative clustering algorithm to the problem of unsupervised learning of clusters of *coefficient vectors* for two image models that correspond to image segments. The first image model is a polynomial form for the observed image measurements; the assumption here is that the image is a collection of several adjoining graph surfaces which are sampled on the raster grid to produce the observed image. The algorithm

proceeds by obtaining vectors of coefficients of least-squares fits to the data in M disjoint image windows. An agglomerative clustering algorithm merges (at each step) the two clusters which yield a minimum global between-cluster Mahalanobis distance. The same framework was applied to segmentation of textured images, but for such images the polynomial model was inappropriate and a parameterized Markov Random Field model was assumed instead.

Wu and Leahy [32] applied the principles of network flow to unsupervised classification and produced a novel hierarchical algorithm for clustering. In essence, the technique views the unlabeled patterns as nodes in a graph, where the weight of an edge (*i.e.*, its capacity) is a measure of similarity between the corresponding nodes. Clusters are identified by removing edges from the graph to produce connected disjoint subgraphs. In image segmentation, pixels which are 4-neighbors or 8-neighbors in the image plane share edges in the constructed adjacency graph, and the weight of a graph edge is based on the strength of a hypothesized image edge between the pixels involved (this strength is calculated using simple derivative masks). Hence, this segmenter works by finding closed contours in the image and is best labeled edge-based rather than region-based.

Vinod *et al.* [33] developed two neural networks which in combination can be used to perform pattern clustering. A two-layer network operates on a multidimensional histogram of the data to identify 'prototypes' which are used to classify the input patterns into clusters. These prototypes are fed to the classification network, another two-layer network operating on the histogram of the input data, but trained to have differing weights from the prototype selection network. In both networks, the histogram of the image is used to weight the contributions of patterns neighboring the pattern under consideration to the location of prototypes or the ultimate classification; as such, it is likely to be more robust compared to techniques which assume an underlying parametric density function for the pattern classes. This architecture was tested on gray-scale and color segmentation problems.

Zhang and Modestino [34] address the *cluster validation* problem in the context of image segmentation. The correct number of clusters (image segments) is decided using an information-theoretic criterion originally developed for the selection of the proper order and parameters of an autoregressive model for time-series data. This criterion is computed from a maximum likelihood estimate of the model parameters, which in this case were obtained from "homogeneous" regions produced by a K-means clustering of the input data. The proper number of clusters is chosen by iterating K from 1 to a predefined maximum K_{max} and selecting that K which minimizes the criterion. This technique was applied to the segmentation of images assumed to contain regions well approximated by Gaussian random fields. More recently, Zhang *et al.* [60] applied the expectation maximization algorithm to the segmentation of images described by several different models.

Jolion *et al.* [35] extracted clusters sequentially from the input pattern set by identifying hyperellipsoidal regions (bounded by loci of constant Mahalanobis distance) which contain a specified fraction of the unclassified points in the set. The extracted regions are compared against the best-fitting multivariate Gaussian density through a Kolmogorov-Smirnov test, and the fit quality is used as a figure of merit for selecting the 'best' region at each iteration. The process continues until a stopping criterion is satisfied. This procedure was applied to the problems of threshold selection for multithreshold segmentation of intensity imagery and segmentation of range imagery.

4.2: Range Image Segmentation

Clustering techniques have been successfully used for the segmentation of range images, which are a popular source of input data for three-dimensional object recognition systems [13]. Range sensors typically return raster images with the measured value at each pixel being the coordinates of a 3D location in space. Depending on the sensor's configuration, these 3D positions can be understood as the locations where rays emerging from the image plane locations in either a parallel bundle or a perspective cone intersect the objects in front of the sensor.

The local feature clustering concept is particularly attractive for range image segmentation since (unlike intensity measurements) the measurements at each pixel have the same units (length): this would make *ad hoc* transformations or normalizations of the image features unnecessary if their goal is to impose equal scaling on those features. However, range image segmenters often add additional measurements to the feature space, removing this advantage.

Hoffman and Jain [38] described a range image segmentation procedure employing squared-error clustering in a six-dimensional feature space as a source of an "initial" segmentation which is refined (typically by merging segments) into the output segmentation. The procedure was enhanced by Flynn and Jain [41] and used in a recent systematic comparison of range image segmenters [42]; as such, it is probably one of the longest-lived range segmenters which has performed well on a large variety of range images.

This segmenter works as follows. At each pixel (i, j) in the input range image, the corresponding 3D measurement is denoted (x_{ij}, y_{ij}, z_{ij}), where typically x_{ij} is a linear function of j (the column number) and y_{ij} is a linear function of i (the row number). A $k \times k$ neighborhood of (i, j) is used to estimate the 3D surface normal $\mathbf{n}_{ij} = (n^x_{ij}, n^y_{ij}, n^z_{ij})$ at (i, j), typically by finding the least-squares planar fit to the 3D points in the neighborhood. The feature vector for the pixel at (i, j) is the six-dimensional measurement $(x_{ij}, y_{ij}, z_{ij}, n^x_{ij}, n^y_{ij}, n^z_{ij})$, and a candidate segmentation is found by clustering these feature vectors. For practical reasons, not every pixel's feature vector is used in the clustering procedure; typically 1000 feature vectors are chosen by structured or random subsampling (Jain and Hoffman concluded that there was little reason to prefer one sampling technique over the other). Recent work by Judd *et al.* [45] demonstrates that a combination of algorithmic enhancements to the clustering algorithm and distribution of the computations over a network of workstations can allow an entire 512×512 image to be clustered in a few minutes.

The issue of interpattern proximity in this six-dimensional feature space deserves some comments. As noted above, the feature space used here is 'mixed' in that three of the features are ratio-scaled continuous measurements (x, y, z), while the remaining features are ordinal-scaled continuous features measured on the unit sphere (n^x, n^y, n^z). The position and orientation features are not directly comparable, and a pragmatic approach was used to make the computation of proximity between patterns more appropriate. Each of the six features was normalized to zero mean and unit standard deviation before the clustering procedure was executed, and the Euclidean distance measure was used. This set of choices works well in practice.

The CLUSTER algorithm was used to obtain segment labels for each pixel. Hoffman and Jain also experimented with other clustering techniques (*e.g.*, complete-link, single-link, graph-theoretic, and other squared-error algorithms) and found CLUSTER to provide the best combination of performance and accuracy. An additional advantage of CLUSTER is that it produces a sequence of output clusterings (*i.e.*, a 2-cluster solution up through a K_{max}-

cluster solution where K_{max} is specified by the user and is typically 20 or so); each clustering in this sequence yields a clustering statistic which combines between-cluster separation and within-cluster scatter. That clustering which optimizes this statistic is chosen as the best clustering.

Each pixel in the range image is assigned the segment label of the nearest cluster center. This minimum distance classification step is not guaranteed to produce segments which are connected in the image plane; therefore, a connected components labeling algorithm allocates new labels for disjoint regions that were placed in the same cluster. Subsequent operations include surface type tests, merging of adjacent patches using a test for the presence of crease or jump edges between adjacent segments, and surface parameter estimation

Figure 5 shows this processing applied to a range image. Part (a) of the figure shows the input range image; part (b) shows the distribution of surface normals. In part (c), the initial segmentation returned by CLUSTER and modified to guarantee connected segments is shown. Part (d) shows the final segmentation produced by merging adjacent patches which do not have a significant crease edge between them. The final clusters reasonably represent distinct surfaces present in this complex object.

4.3: Texture Image Segmentation

The analysis of textured images has been of interest to researchers for several years. Tuceryan and Jain [46] provide a comprehensive survey of texture definitions, models, and analysis techniques. Texture segmentation techniques have been developed using a variety of texture models and image operations. In this section, we survey some clustering-based approaches to the segmentation of texture images.

Nguyen and Cohen [31] addressed the texture image segmentation problem by modeling the image as a hierarchy of two Markov Random Fields, obtaining some simple statistics from the image texture in each block to form a feature vector, and clustering these blocks using a fuzzy K-means clustering method. The clustering procedure here is modified to jointly estimate the number of clusters as well as the fuzzy membership of each feature vector to the various clusters.

Jain and Farrokhnia [43] developed a system for segmenting texture images. The system uses Gabor filters to obtain a set of 28 orientation- and scale-selective features characterizing the texture in the neighborhood of each pixel. To avoid the "curse of dimensionality" problem associated with a large number of features per pattern, the set of features (ordered by sample variance) which collectively capture a prespecified fraction of the total variance of the original 28 features is used as a reduced feature set. A saturating nonlinear transformation is applied independently to each of the retained feature images, and the feature images are filtered with a uniform kernel. These filtered feature images are then subsampled uniformly to select 1000 or fewer feature vectors, which are then clustered with the CLUSTER program described above. An index statistic proposed by Dubes [44] is used to select the best clustering. Minimum distance classification is used to label each of the original image pixels. This technique was tested on several texture mosaics including the natural Brodatz textures and synthetic images. Figure 6(a) shows an input texture mosaic consisting of four of the popular Brodatz textures [16]. Part (b) shows the segmentation produced when the Gabor filter features are augmented to contain spatial information (pixel coordinates). This Gabor filter technique has proven very powerful and has been extended to the automatic segmentation of text in documents [40] and segmentation of objects in complex background [39].

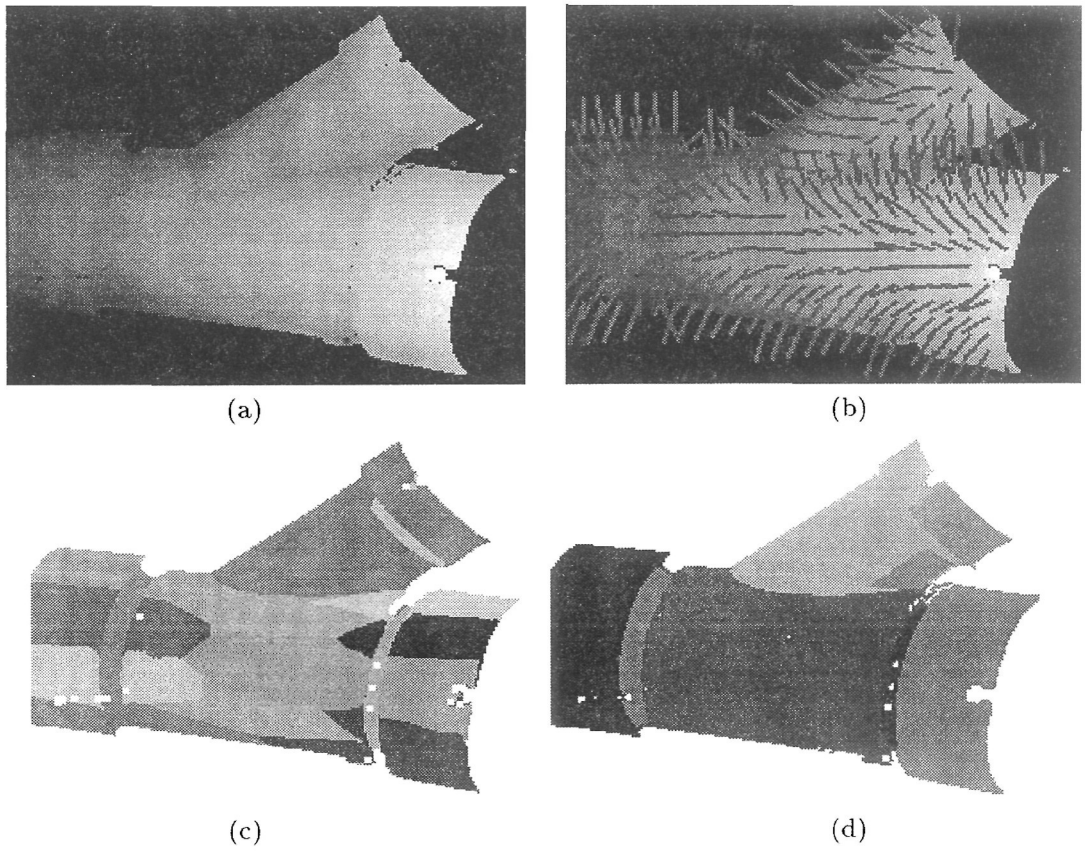
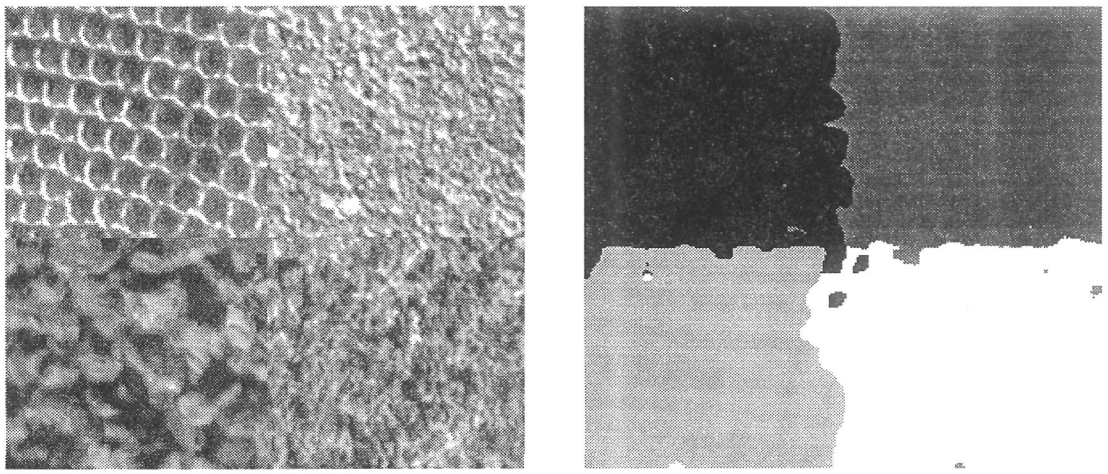


Figure 5. Range image segmentation using clustering. (a): Input range image. (b): Surface normals for selected image pixels. (c): Initial segmentation (19 cluster solution) returned by CLUSTER using 1000 six-dimensional samples from the image as a pattern set. (d): Final segmentation (8 segments) produced by postprocessing.

Mao and Jain [56] combined multiresolution analysis and autoregressive image models in a texture classification and segmentation system. The autoregressive image model is rotation-invariant and admits a simple parameter estimation procedure. Model parameters are computed at multiple image resolutions (constructed using low-pass filtering and subsampling). The performance of this system as a classifier (*i.e.*, identifying input textures as belonging to one of several predefined classes characterized by model parameters) was impressive. The system was modified for use as a texture segmenter. Instead of subsampling the image, multiple parameter estimation windows of different sizes were designed and used, and a squared-error clustering algorithm was used to generate segmentations. A careful evaluation of the ability of features to discriminate between clusters, the assignment of feature weights, and the evaluation of clustering output are notable features of this work.



(a) (b)

Figure 6. Texture image segmentation results. (a): Four-class texture mosaic. (b): Four-cluster solution produced by CLUSTER with pixel coordinates included in the feature set.

4.4: Color/Multispectral Image Segmentation

Haralick and Kelly [26] proposed two clustering procedures for images (including multispectral images). The *spatial* clustering procedure sequentially builds an image partition starting with a single cluster; at each iteration, a new cluster is defined as a connected subset of pixels with the same measurement value(s), and grown to connected pixels with similar measurement value(s). A *measurement space* clustering procedure iteratively forms connected regions of the measurement space using a probabilistic measure of proximity between subsets in that space. The latter procedure was tested on an aerial image of an urban scene with three measurements per pixel.

Amadasun and King [25] constructed a segmentation system for multispectral imagery employing agglomerative clustering. Image neighborhoods which are uniform according to a mean-value criterion are extracted, summarized by their mean vectors, and fed to an agglomerative clustering procedure to identify different segments. The agglomerative procedure is close to a single-link algorithm, merging the classes of the two closest patterns until the number of output classes equals the (prespecified) number of desired segments. The final image segmentation is obtained from a minimum-distance classification of each (multispectral) pixel with respect to the obtained clusters.

Coleman and Andrews [28] applied a K-means clustering procedure to multispectral image segmentation. Their presentation is rich with the essential and practical details of applying pattern recognition methodology to the image segmentation problem (*e.g.*, the necessity of an eigenvector rotation to obtain decorrelated features, the use of the Bhattacharyya distance measure for feature selection). A variety of features were used in order to classify images with varying amounts of texture; these features included the responses of the Sobel operator to local intensity as well as nearly uniform regions of varying sizes obtained from a nonlinear image filter. The K-means procedure was executed on a subset of the output features

obtained by eigenvector rotation and selection, and output clusterings consisting of two to sixteen clusters were produced (beginning with two clusters, a new cluster is formed at each iteration by identifying the most outlying pattern and using it as the new cluster center). A final image segmentation is obtained using minimum distance classification with the set of obtained feature vectors.

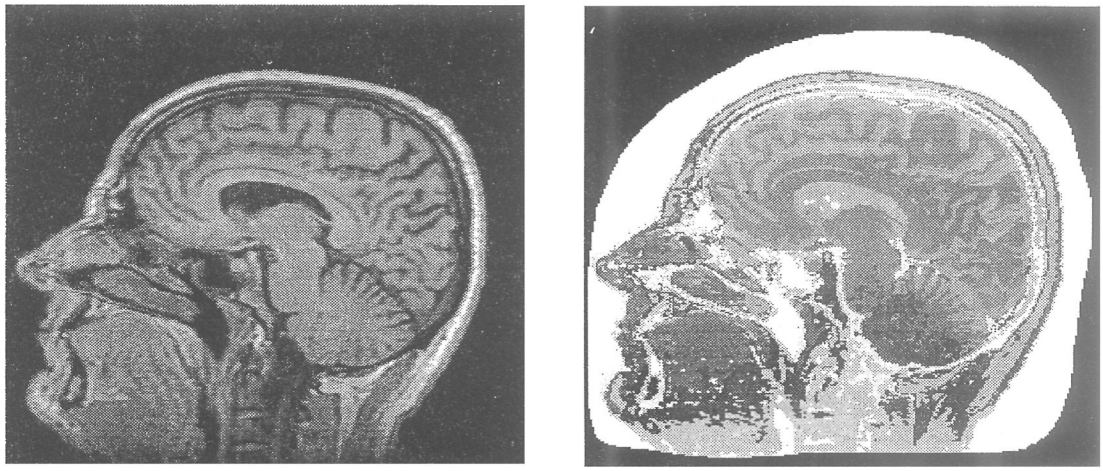
Gowda [30] applied the ISODATA [14] clustering algorithm to the segmentation of multispectral images (LANDSAT frames). A notable feature of this method was the use of several transformations to take the original four-dimensional measurement at each pixel into the HSV (hue, saturation, value) color basis, and from there into a pseudo RGB (red, green, blue) basis. The 3D patterns in the RGB space are condensed into a 2D array which summarizes patterns within a small distance of one another by their mean vectors. This set of mean vectors is then clustered using a multistage variant of the popular ISODATA algorithm, and the output segmentation is obtained by a minimum distance classification of the input pixels with respect to the derived cluster centers. This approach to image segmentation uses very little memory due to the condensing procedure.

Bryant [47] developed a customized clustering algorithm for the segmentation of LANDSAT image data. This work is notable for its clear description of a system and its underlying algorithms, and for its comprehensive survey and critique of clustering-based multispectral image segmentation techniques *circa* 1979. Bryant's system has two major phases. In the first phase, simple gradient thresholding is used to identify 'fields' (small uniform regions) in the input data. Samples from these fields are used to form both the subset of data fed to the clustering algorithm and that algorithm's initial set of candidate cluster centers. The clustering algorithm itself works by performing minimum-distance classification with rejection of pixels which are too far from any cluster center. Image-plane cleaning operations (designed to label pixels containing a mixture of class information) are also applied to improve the results. The number of clusters can be automatically adjusted by internally judging the quality of the current set of cluster centers. This approach was tested on several LANDSAT scenes.

Fukuda [29] developed two clustering procedures for segmentation and applied them to color images. The first procedure segments the image by recursively subdividing the image into blocks until each block has a small dispersion about its sample mean in the measurement space, and then grouping blocks with similar mean vectors into connected segments. The second procedure performs one subdivision step (without recursion), obtains a sequence of clusterings by varying a dispersion threshold used as a merging criterion, selects that threshold which yielded the maximum number of clusters, summarizes each cluster by a cluster center, and merges those cluster centers to yield fewer than a prespecified number. In each algorithm, the final image segmentation is employed by a minimum-distance classification of each pixel with respect to the set of output cluster centers.

Uchiyama and Arbib [37] applied competitive learning techniques to the color image segmentation problem. The authors demonstrate an equivalence between clustering and vector quantization, and 'units' which represent cluster centers are sequentially generated. After each generation, the units compete for randomly chosen members of the input pattern set, and the winning unit (which is closest to the pattern) has its weight vector (*i.e.*, the location of the cluster center) updated toward the pattern. Units are created where the local density of patterns is large as evidenced by a large number of 'wins' by a specific unit. In applying this technique to color image segmentation, the authors motivate the use of a feature transformation to improve the segmentation results.

Clustering can be used as a preprocessing stage to identify pattern classes for subsequent



(a) (b)

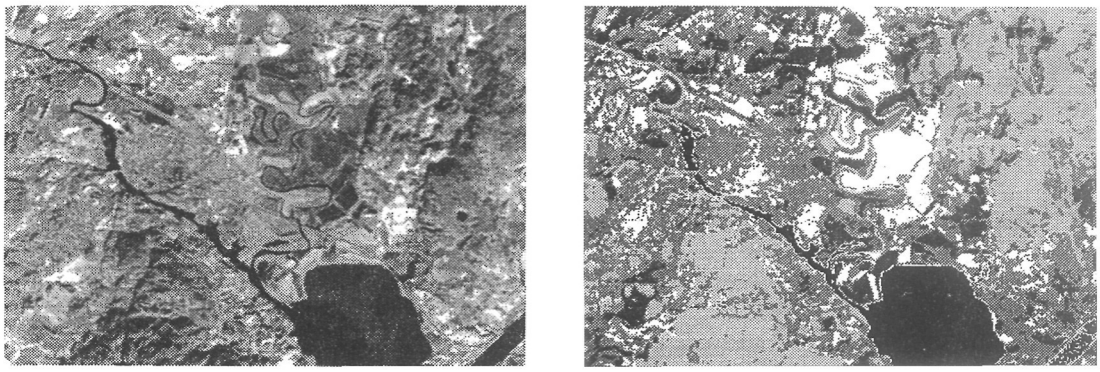
Figure 7. Multispectral Medical Image Segmentation. (a): A single channel of the input image. (b): 9-cluster segmentation.

supervised classification. Taxt and Lundervold [53, 54] employed a partitional clustering algorithm and a manual labeling technique to identify material classes (*e.g.*, cerebrospinal fluid, white matter, striated muscle, tumor) in registered images of a human head imaged at five different magnetic resonance imaging channels (yielding a five-dimensional feature vector at each pixel). A number of clusterings were obtained and combined with domain knowledge (human expertise) to identify the different classes. Decision rules for supervised classification were based on these obtained classes. Figure 7(a) shows one channel of an input multispectral image; part (b) shows the 9-cluster result.

Solberg [62] applied the K-means algorithm to segmentation of LANDSAT imagery. Initial cluster centers were chosen interactively by a trained operator, and correspond to land-use classes such as urban areas, soil (vegetation-free) areas, forest, grassland, and water. Figure 8(a) shows the input image rendered as grayscale; part (b) shows the result of the clustering procedure.

4.5: Multiple Motion Segmentation

Cluster analysis has also been applied to the problem of resolving multiple motions in image sequences. The intuitive model used in current work is that the observed image is composed of *layers*, each with an associated motion, and the segmentation problem is expected to resolve the layers. Jepson and Black [59] used the expectation maximization algorithm [17] to resolve a mixture density model for the motion parameters. Here, the mixture models the presence of several different motion classes. Sawhney *et al.* [57] adopt a similar approach. Wang and Adelson [58] used a K-means clustering procedure to segment different affine motion classes observed in image sequences. Dubuisson and Jain [61] fused motion segmentation, color segmentation based on a split-and-merge paradigm and edge information from the Canny edge detector to extract the contours of moving objects. Figure 9



(a) (b)

Figure 8. LANDSAT image segmentation (a): Original image (©ESA/EURIMAGE/Sattelitbild). (b): Clustered scene.

shows the segmentation obtained in a single frame of a motion sequence using this technique.

5: Summary

In this chapter, the application of clustering methodology to image segmentation problems has been motivated and surveyed. The historical record shows that clustering is a powerful tool for obtaining classifications of image pixels. Key issues in the design of any clustering-based segmenter are the choice of pixel measurements (features) and dimensionality of the feature vector (*i.e.*, should the feature vector contain intensities, pixel positions, model parameters, filter outputs, *etc*?), a measure of similarity which is appropriate for the selected features and the application domain, the identification of a clustering algorithm (*e.g.*, squared-error, mode-seeking, graph-theoretic, *etc.*), the development of strategies for feature and data reduction (to avoid the "curse of dimensionality" and the computational burden of classifying large numbers of patterns), and the identification of necessary pre- and post-processing techniques (*e.g.*, image smoothing and minimum distance classification). The use of clustering for segmentation dates back to the 1960s and new variations continue to emerge in the literature. Challenges to the successful use of clustering include the high computational complexity of many clustering algorithms and their incorporation of strong assumptions (often multivariate Gaussian) about the multidimensional shape of clusters to be obtained. The ability of new clustering procedures to handle concepts and semantics in classification (in addition to numerical measurements) will be important for certain applications [52, 51]; we see opportunities here for the adoption of ideas arising in the machine learning literature. With its rich history, our reasonable understanding of the properties and limitations of clustering methodology, and the prospect of increasingly powerful computer systems to use in image analysis systems, we see potential for continued contributions to this fruitful research area.

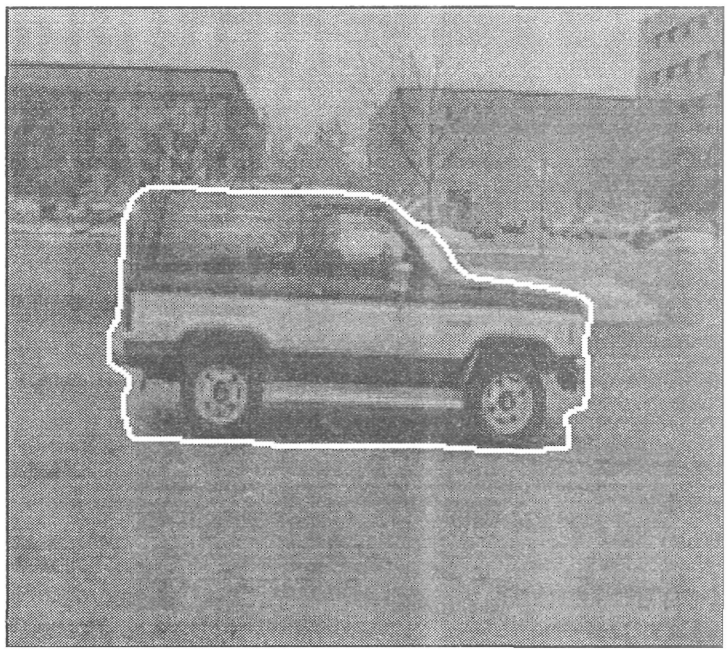


Figure 9. Segmentation of a vehicle's contour from fused motion, color, and edge information.

6: Acknowledgements and Dedication

We thank Jianchang Mao, Torfinn Taxt, Anne Schistad Solberg, Anne Marie Fenstad, Marie-Pierre Dubuisson Jolly and Yu Zhong for their contributions to this chapter. Our research in clustering methodology and its applications in computer vision has been supported over the past 20 years by the National Science Foundation, NASA, Michigan State University, Washington State University, and a number of industrial organizations. We dedicate this chapter to the memory of Dr. Richard C. Dubes (1934-1993): a pioneering researcher in clustering, collaborator, teacher, mentor, and friend.

References

[1] A. Rosenfeld and A.C. Kak, *Digital Picture Processing* (2nd ed.), Academic Press, 1982.

[2] A. Rosenfeld, M.K. Huang and V.B. Schneider, "An Application of Cluster Detection to Text and Picture Processing," *IEEE Trans. on Information Theory* 15(6):672–681, 1969.

[3] C.A. Sher and A. Rosenfeld, "Pyramid Cluster Detection and Delineation by Consensus," *Pattern Recognition Letters* 12(8):477–482, 1991.

[4] S. Dunn, L. Janos and A. Rosenfeld, "Bimean Clustering," *Pattern Recognition Letters* 1:169–173, 1983.

[5] B.J. Schachter, L.S. Davis and A. Rosenfeld, "Some Experiments in Image Segmentation by Clustering of Local Feature Values," *Pattern Recognition* 11:19–28, 1979.

[6] J.-M. Jolion and A. Rosenfeld, "Cluster Detection in Background Noise," *Pattern Recognition* 22(5):603–607, 1989.

[7] A. Scher, M. Shneier and A. Rosenfeld, "Clustering of Collinear Line Segments," *Pattern Recognition* 15(2):85–91, 1982.

[8] A. Rosenfeld and Y.H. Lee, "A Clustering Heuristic for Line-Drawing Analysis," *IEEE Trans. on Computers* C-21:904–911, 1972.

[9] F.R.D. Velasco and A. Rosenfeld, "Some Methods for the Analysis of Sharply Bounded Clusters," *IEEE Trans. on Systems, Man, and Cybernetics* SMC-10(8):511–518, 1980.

[10] T. Pavlidis, *Algorithms for Graphics and Image Processing*, Computer Science Press, 1982.

[11] R. Jain, R. Kasturi and B.G. Schunck, *Machine Vision*, McGraw-Hill, 1995.

[12] A.K. Jain and R.C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, 1988.

[13] A.K. Jain and P.J. Flynn (eds.), *Three Dimensional Object Recognition Systems*, Elsevier, 1993.

[14] R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, 1973.

[15] E. Backer, *Computer-Assisted Reasoning in Cluster Analysis*, Prentice–Hell, 1995.

[16] P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover, 1966.

[17] G.J. McLachlan and K.E. Basford, *Mixture Models: Inference and Applications to Clustering*, Marcel Dekker, 1988.

[18] F. Farrokhnia and A.K. Jain, "A Multi-channel Filtering Approach to Texture Segmentation," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR '91)*, Maui, 364–370, 1991.

[19] S. Nadabar and A.K. Jain, "MRF Model-Based Segmentation of Range Images," *Proc. Third International Conf. on Computer Vision*, Osaka, 667–671, 1990.

[20] R. Ohlander, K. Price and D.R. Reddy, "Picture Segmentation Using a Recursive Region Splitting Method," *Computer Graphics and Image Processing* 8(3):313–333, 1979.

[21] K.S. Fu and J.K. Mui, "A Survey on Image Segmentation," *Pattern Recognition* 13:3–16, 1981.

[22] J. Weszka, "A Survey of Threshold Selection Techniques," *Pattern Recognition* 7:259–265, 1978.

[23] O.D. Trier and A.K. Jain, "Goal-Directed Evaluation of Binarization Methods," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 17(12):1191–1201, 1995.

[24] R.C. Dubes and A.K. Jain, "Clustering Techniques: The User's Dilemma," *Pattern Recognition* 8:247–260, 1976.

[25] M. Amadasun and R.A. King, "Low-Level Segmentation of Multispectral Images via Agglomerative Clustering of Uniform Neighborhoods," *Pattern Recognition* 21(3):261–268, 1988.

[26] R.M. Haralick and G.L. Kelly, "Pattern Recognition with Measurement Space and Spatial Clustering for Multiple Images," *Proc. IEEE* 57(4):654–665, 1969.

[27] J.F. Silverman and D.B. Cooper, "Bayesian Clustering for Unsupervised Estimation of Surface and Texture Models," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 10(4):482–495, 1988.

[28] G.B. Coleman and H.C. Andrews, "Image Segmentation by Clustering," *Proc. IEEE* 67(5):773–785, 1979.

[29] Y. Fukuda, "Spatial Clustering Procedures for Region Analysis," *Pattern Recognition* 12:395–403, 1980.

[30] K.C. Gowda, "A Feature Reduction and Unsupervised Classification Algorithm for Multispectral Data," *Pattern Recognition* 17(6):667–676, 1984.

[31] H.H. Nguyen and P. Cohen, "Gibbs Random Fields, Fuzzy Clustering, and the Unsupervised Segmentation of Textured Images," *CVGIP: Graphical Models and Image Processing* 55(1):1–19, 1993.

[32] Z. Wu and R. Leahy, "An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image Segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 15(11):1101–1113, 1993.

[33] V.V. Vinod, S. Chaudhury, J. Mukherjee and S. Ghose, "A Connectionist Approach for Clustering with Applications in Image Analysis," *IEEE Trans. on Systems, Man, and Cybernetics* 24(3):365–384, 1994.

[34] J. Zhang and J.W. Modestino, "A Model-Fitting Approach to Cluster Validation with Application to Stochastic Model-Based Image Segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 12(10):1009–1017, 1990.

[35] J.-M. Jolion, P. Meer, and S. Bataouche, "Robust Clustering with Applications in Computer Vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 13(8):791–802, 1991.

[36] Y.W. Lim and S.U. Lee, "On the Color Image Segmentation Based on the Thresholding and the Fuzzy c-Means Techniques," *Pattern Recognition* 23(9):935–952, 1990.

[37] T. Uchiyama and M. Arbib, "Color Image Segmentation Using Competitive Learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 16(12):1197–1206, 1994.

[38] R.L. Hoffman and A.K. Jain, "Segmentation and Classification of Range Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence* PAMI-9(5):608–620, 1987.

[39] A.K. Jain, N.K. Ratha and S. Lakshmanan, "Object Detection using Gabor Filters," *Pattern Recognition*, to appear in 1996.

[40] A.K. Jain and S. Bhattacharjee, "Text Segmentation using Gabor Filters for Automatic Document Processing," *Machine Vision and Applications* 5:169–184, 1992.

[41] P.J. Flynn and A.K. Jain, "BONSAI: 3D Object Recognition Using Constrained Search," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 13(10):1066–1075, 1991.

[42] A. Hoover, G. Jean-Baptiste, X. Jiang, P.J. Flynn, H. Bunke, D. Goldgof, K. Bowyer, D. Eggert, A. Fitzgibbon and R. Fisher, "An Experimental Comparison of Range Image Segmentation Algorithms," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, to appear.

[43] A.K. Jain and F. Farrokhnia, "Unsupervised Texture Segmentation Using Gabor Filters," *Pattern Recognition* 24(12):1167–1186, 1991.

[44] R.C. Dubes, "How Many Clusters are Best? – An Experiment," *Pattern Recognition* 20:645–663, 1987.

[45] D. Judd, P. McKinley and A.K. Jain, "Large-Scale Parallel Data Clustering," *Int. Conf. on Pattern Recognition*, under review.

[46] M. Tuceryan and A.K. Jain, "Texture Analysis," in *Handbook of Pattern Recognition and Computer Vision* (C. Chen, L. Pau and P. Wang (eds.)), World Scientific, 1993, pp. 235–276.

[47] J. Bryant, "On the Clustering of Multidimensional Pictorial Data," *Pattern Recognition* 11:115–125, 1979.

[48] J. Mao and A.K. Jain, "A Self-Organizing Network for Hyperellipsoidal Clustering (HEC)," *IEEE Trans. on Neural Networks* 7(1):16–29, 1996.

[49] C.T. Zahn, "Graph-theoretical Methods for Detecting and Describing Gestalt Clusters," *IEEE Trans. on Computers* C-20:68–86, 1971.

[50] N. Ahuja and M. Tuceryan, "Extraction of Early Perceptual Structure in Dot Patterns: Integrating Region, Boundary, and Component Gestalt," *Computer Vision, Graphics, and Image Processing* 48:304–356, 1989.

[51] M.N. Murty and A.K. Jain, "Knowledge-Based Clustering Scheme for Collection Management and Retrieval of Library Books," *Pattern Recognition* 28(7):949–964, 1995.

[52] R.S. Michalski and R. Stepp, "Automatic Construction of Classifications: Conceptual Clustering *versus* Numerical Taxonomy," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 5:396–409, 1983.

[53] T. Taxt and A. Lundervold, "Multispectral Analysis of the Brain Using Magnetic Resonance Imaging," *IEEE Trans. on Medical Imaging* 13(3):470–481, 1994.

[54] A. Lundervold, A.M. Fenstad, L. Ersland and T. Taxt, "Brain Tissue Volumes from Multispectral 3D MRI–A Comparative Study of Four Classifiers," *Proc. Society of Magnetic Resonance* (fourth meeting), New York, p. 33, 1996.

[55] R.A. Redner and H.F. Walker, "Mixture Densities, Maximum Likelihood and the EM Algorithm," *SIAM Review* 26(4):195–239, 1984.

[56] J. Mao and A.K. Jain, "Texture Classification and Segmentation Using Multiresolution Simultaneous Autoregressive Models," *Pattern Recognition* 25(2):173–188, 1992.

[57] H. Sawhney, S. Ayer and M. Gorkani, "Dominant and Multiple Motion Estimation for Video Representation," *Proc. International Conf. on Image Processing (ICIP '95)*, Washington DC, vol. 1, 322–325, 1995.

[58] J.Y.A. Wang and E.H. Adelson, "Layered Representation for Motion Analysis," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR '93)*, New York, 361–366, 1993.

[59] A. Jepson and M.J. Black, "Mixture Models for Optical Flow Computation," *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR '93)*, New York, 760–761, 1993.

[60] J. Zhang, J.W. Modestino and D.A. Langan, "Maximum-Likelihood Parameter Estimation for Unsupervised Stochastic Model-Based Image Segmentation," *IEEE Trans. on Image Processing* 3(4):404–420, 1994.

[61] M.P. Dubuisson and A.K. Jain, "Contour Extraction of Moving Objects in Complex Outdoor Scenes," *International J. of Computer Vision* 14:83–105, 1995.

[62] A.H. Schistad Solberg, "Data Fusion and Texture in Classification of Synthetic Aperture Radar Imagery," Ph.D. thesis, Department of Informatics, University of Oslo, June 1995.

Boundary Encoding Revisited

Herbert Freeman
Department of Electrical and Computer Engineering
Rutgers University, Piscataway, NJ 08855-0909

Abstract

In 1969, at a workshop organized by Azriel Rosenfeld, a paper was presented dealing with boundary encoding and processing. In the intervening years, many advances and extensions of the techniques presented then have been made. This paper reviews some of these developments, with particular emphasis on various generalizations of the chain coding scheme.

1: Introduction

The processing of two-dimensional shape is a topic of abiding interest to those concerned with image processing, computer vision, and pattern recognition. A two-dimensional shape, familiarly often referred to as a "blob," is fully defined by its bounding contour. This allows us to convert a problem dealing with the shape of a region into a more tractable one-dimensional one dealing with a closed contour. Why do we want to process a blob? There are a variety of reasons. We may want to analyze it, that is, determine its area, its perimeter, its maximum diameter, whether or not it is convex, the location of its centroid, whether or not it has one or more axes of symmetry, etc. Or we may want to establish the degree of similarity it has with another blob, either with or without regard to size or orientation. Finally we may want to classify it on the basis of one or more of its shape characteristics.

In 1970 the author published a paper in a book co-edited by Bernice Lipkin and Azriel Rosenfeld under the title of *Boundary Encoding and Processing* [1]. The paper addressed the problem of describing a blob's boundary lines, noted the distinctions among the geometric boundary, "black" or interior boundary, and "white" or exterior boundary, and pointed out how sets of directed contours can be used to describe graytone blobs. The chain-coding technique was demonstrated in the context of multi-level graytone blobs and extended to the direct quantization and encoding of boundary contours, whether open or closed. A variety of processing and manipulation algorithms for chain-encoded curves (or boundary contours) were described, and the now well-known three properties of a digital straight line were postulated. It appears fitting that today, 26 years later, as we honor Professor Rosenfeld's many years of contributions to the field of image processing and its allied sciences, the author should

0-8186-7644-2 $5.00

re-visit the subject of boundary encoding and describe what has happened in the interval, where we have progressed, and where difficult problems still challenge us.

2: A Blob and Its Boundary

When we speak of a blob, we think of a closed, homogeneous region, simply- or multiply-connected, which models some physical entity, e.g., an area feature on a map (state, county, lake), a blood cell viewed through a microscope, or a defect in an otherwise clear sheet of glass. If the blob corresponds to something in the physical world, it must be digitized before it can be processed by computer. With shape as our only concern, we have the choice of either digitizing an image of the blob and then extracting the boundary from the resulting pixel array, or digitizing the boundary directly. Depending on which approach we follow, somewhat different problem situations are encountered.

2.1: Extracting a contour from a digitized blob

If we first digitize the blob and then look for its boundary, we find that there are at least three different ways of defining the boundary. We can use the contour that literally bounds the connected set of pixels forming the blob, and obtain what in Fig. 1 is marked as the *geometric boundary* (also sometimes referred to as the *crack boundary* [2]). Or we can connect the centers of bounding pixels - either those inside the blob or those outside, obtaining, as shown in Fig. 1, the *interior boundary* in the former case and the *exterior boundary* in the latter. As was proved in [1], the interior boundary will always be shorter than the geometric boundary, and the exterior boundary will be shorter if the geometric boundary consists of 10 or more distinct straight-line segments. All three are functions of the digitization process, as well as of any smoothing and thresholding that was applied, and none can be said a priori to be a more faithful rendition of the original analog contour than another.

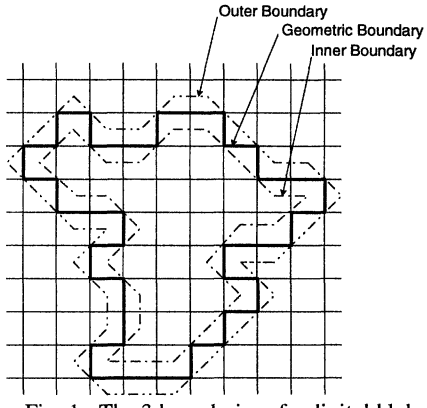


Fig. 1. The 3 boundaries of a digital blob.

During the last few years some researchers have introduced additional ways of describing a digital blob's contour. One interesting variant is the so-called mid-crack scheme [3, 4], which uses straight-line segments of length 1 or $\sqrt{2}/2$ (times the size of a pixel edge) to connect the mid-points of bounding edges. It leads to a bounding contour that more closely "hugs" the geometric boundary but involves more segments than either the interior or exterior boundaries, as illustrated in Fig. 2. Whether or not it has advantages over the more straight-forward interior or exterior boundaries, which use segments of length 1 and $\sqrt{2}$ has not been established. Also, as we shall

show later, since precision is solely a function of the underlying quantization lattice, but processing performance is a function of the number of segments, it may be preferable to look for boundary representations that use the fewest possible segments.

2.2: Direct digitization of a contour

The alternate way to obtain the digital representation of a blob's bounding contour is to trace it in the analog domain and digitize it directly. For a multiply-connected blob, we would trace the exterior boundary in the clockwise sense, and any "holes" in a clockwise sense, thus assuring that the interior will always be to the right. The process of digitization is one of description, and description implies quantization [5]. Quantization is defined in term of *form*, the *size* of the quanta, and the *approximant* used to represent the quanta. Many tradeoffs are possible among these, and the choices made can affect the performance of the processing to be carried out on the resulting digital data.

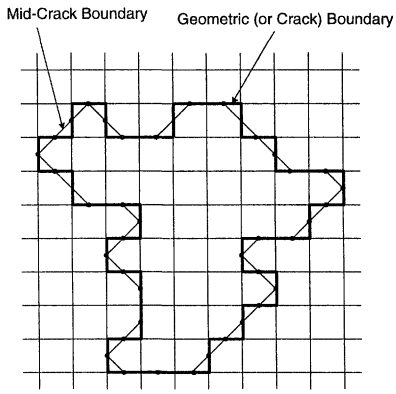


Fig. 2. The mid-crack boundary.

The most common way of representing a boundary contour is to set up a Cartesian coordinate system and describe the contour in terms of a series of (x, y)-coordinates. The points used to describe the contour will necessarily be nodes of an implied uniform square lattice whose spacing corresponds to a unit change in the least-significant bit position of the x and y coordinates. In general, this built-in precision limitation will be much finer than what can be justified on the basis of the data's underlying physical precision limitations, suggesting the use of a correspondingly larger lattice spacing. This is illustrated in Fig. 3, where the lattice spacings Δx and Δy correspond to the smallest significant change in coordinates. Here the *form* of the quantization is the uniform square lattice, the *size* is the lattice spacing, and the *approximant* is yet to be specified. A node becomes a curve point (a selected node for the digital approximation) when the curve crosses a lattice line within half a lattice spacing of the node [6].

If the lattice spacing is chosen as indicated, no useful (i.e., "no significant") information about the curve exists between two adjacent nodes. It follows that to obtain the digital representation of the contour, we should join successive nodes with only the most primitive approximant, namely, a short straight-line segment. This then yields the digital approximation or *chain* representation of a curve, where for each selected curve point the next in sequence is one of the eight lattice points surrounding it, connected with a segment of length 1 or $\sqrt{2}$ (times the lattice spacing) [6, 7].

It has long been known that if a contour is to be approximated by a sequence of straight-line segments, the segments should be short where the curvature is great, and should be correspondingly longer where the curvature is slight. This suggests that we represent a contour by means of line segments of varying length, selected so as to keep some error measure within a specified bound. The error measure could be the maximum distance between the curve and the approximating line segment, the average distance, the mean-square distance, or the in-between area. The result will be a polygonal approximation of the contour. As long as we insist that wherever a line segment crosses one of the lattice lines, the distance from the curve to the nearest node be no greater than half a lattice spacing, we shall retain the full precision set by the lattice.

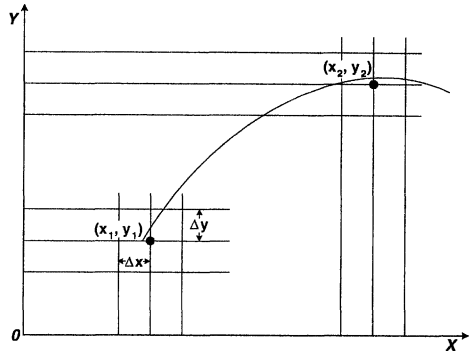


Fig. 3. Quantization of a contour

In Fig. 4 we show a point ***A*** on a square lattice, surrounded by concentric "rings" of nodes. The nodes in the inner-most ring are labeled counterclockwise from 0 to 7; those in the next ring, from 8 to 23, etc. Each ring contains $8n$ nodes, where n is the ring number. The sum of all nodes for rings 1 through n is equal to $4n(n+1)$. Clearly, if we can identify a longer segment, i.e., one that would connect point ***A*** to a node of ring $n > 1$, while satisfying the half lattice-spacing precision requirement, then our approximation will consist of fewer segments. For $n = 1$, we have the basic chain-code, and for rings of all values from 1 to N, where N is such that the contour lies entirely within this ring, we have unrestricted polynomial approximation.

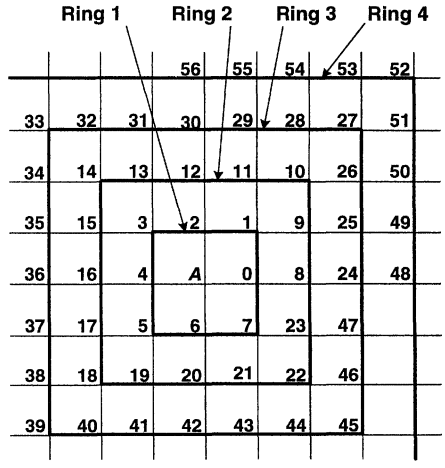


Fig. 4. The node rings surrounding a current node.

A contour-approximation scheme that provides for many line segments that rarely occur will tend to be inefficient to encode. This suggests that we curtail the number of permissible line segments. In the familiar chain-coding scheme, only the nodes in ring 1 of Fig. 4 are permissible "next" nodes. Contour representation based on this scheme has been widely used for more than 30 years; it is simple, efficient, compact, and facilitates the design of simple processing and manipulation algorithms [1, 7]. A detailed analysis of the performance of chain codes can be found in [8]

2.3 The generalized chain representation.

In the late 1970s the *generalized chain coding scheme* was introduced [9,10]. It allows the building of a contour's digital approximation by using nodes from ring 1 plus those from one or more additional concentric rings. Generalized codes have been explored based on the use of rings 1-2, 1-2-3, 1-2-3-4, and larger. And, of course, we can use a ring set in which we skip some intermediate rings, such as ring sets 1-3 and 1-2-5, which contain 32 and 64 nodes, respectively. Note that ring 1 must always be included lest the precision be compromised. Digitization of a contour using the generalized chain concept is analogous to that used for the basic 8-direction chain [7], The procedure always attempts first to use the largest possible line segments (beginning with the largest allowed ring) and then steps down to smaller rings (ultimately to ring 1) if necessary to satisfy the precision requirement [10].

For purposes of numerical encoding, the segments in a ring set are normally assigned sequential numbers, in a counterclockwise spiraling manner. If a ring is skipped, the number are reassigned as needed, e.g., from 0 to 31 for ring set 1-3.

The generalized coding scheme provides an expanded "vocabulary" of line segments for representing a contour, thereby offering the possibility of a smoother representation as well as of fewer approximating segments. Generally, algorithms for analyzing or manipulating such contour representations process them one segment at a time. With the use of appropriate table-lookup schemes, as long as the ring size is kept to a modest value (e.g., 5 or 7), the complexity of processing a segment is virtually independent of the segment's length. Hence the processing time will be proportionately reduced, the fewer the number of segments needed to represent a particular contour [11].

2.4: Use of curved segments as approximants

In general, if the lattice spacing is properly selected to preserve the significant detail of a contour, the likelihood of using segments from a particular ring decreases rapidly with increasing ring number. Thus if is desired further to reduce the number of segments used in a particular approximation, we must relax the requirement that the approximants be straight-line segments. Thus for rings of size 4 or larger, we shall allow, in addition to the straight-line segment, 3 circular arcs of progressively increasing curvature to either side of the straight segment, as illustrated in Fig. 5. This greatly increases the number of available approximants (to 7 x 32 = 224 for ring 4, for example), while simultaneously increasing the probability that such approximants will, in fact, find use [12]. Although there is no gain in precision (as long as we use the same lattice spacing), this generalized "polycurve" representation can yield faster processing and smoother displays.

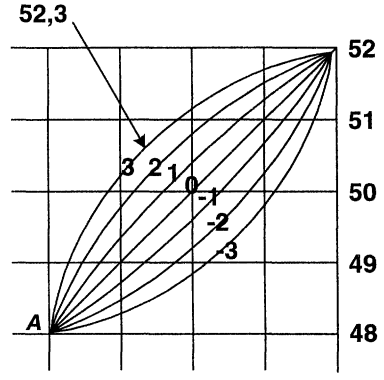
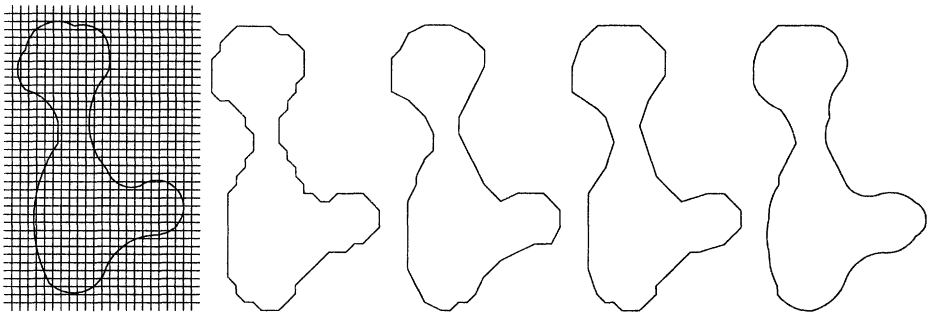


Fig. 5. The set of 7 approximants associated with a node in ring 4.

The effects of the various generalizations on the chain code are illustrated in Fig. 6. The reduction in the number of approximating segments as well as the increasing smoothness are clearly evident, as we go from the basic 8-direction chain representation to the ring 1-3 generalized chain and the ring 1-4 polycurve representation. The number of segments required to represent this particular blob was found to be 87 in (b), 48 in (c), 42 in (d), and 27 in (e).



Fig, 6. (a) A given blob with the overlaid digitizing lattice. (b) Representation using ring 1 only (basic chain code), (c) using rings 1 and 2, (d) using rings 1 and 3, and (e) using a 1-4 ring polycurve.

2.5 Open curves

The techniques described for directly representing the boundary of a blob apply equally when the contour is not closed. However, if we want to apply the techniques to general line drawings, we must allow for junctions. One approach is to represent each segment separately, and then make provision for specifying the starting-point coordinates of each segment, say, by embedding special control codes in each chain-code string [7]. An elegant alternative is provided by the *primitives chain code* introduced by O'Gorman, which directly retains branching and junction topology information [13].

3: Hierarchical Representations

Paralleling the development of techniques for serially processing contours in full description, have been efforts at parallel processing and processing at different levels of resolution. The main motivation for considering parallel processing is the obvious one of reduced processing time. Parallel processing has long been considered for image arrays, but its application to curves (open or closed) and general line drawings had not been given much attention until fairly recently, mainly because of the inherently serial nature of many curve representation schemes.

An effective framework for parallel processing of contours is provided by resorting to multi-resolution extensions of the chain coding scheme [14]. Lattices of different resolution are used, with the finest level corresponding to the precision limit, and the coarsest to a single cell. Developments of this approach have led to the RULI chain code [15] and the chain pyramid [16]. Hierarchical multi-resolution approaches addi-

tionally are of interest for transmitting line-drawing information, as such schemes allow useful tradeoffs between resolution and transmission time, an important current research topic.

4: Shape Characterization

During the 27 years since the paper on boundary encoding and processing appeared, enormous progress has been made in characterizing and analyzing two-dimensional shape. We can define a quantitative measure for shape in terms of the bending energy needed to deform an elastic ring into the desired shape. The more intricate the shape and desired non-circularity, the greater the required bending energy. For an open curve, we follow the same concept but apply it to a straight elastic rod. Note that two curves may have the same energy measure but still be of widely different shape [17].

Of particular interest regarding shape are so-called *critical points*, such as sharp slope discontinuities (i.e., "corners"), points of inflection, points of maximum curvature, junctions, and curve endings, with the latter two occurring, of course, only in the case of open curves. Critical points are particularly attractive in shape matching applications, as they are invariant to scale and rotation, and, at least with respect to the corners, tend to be relatively robust [18-20].

5: Conclusion

A brief overview has been provided of some selected advances in boundary representation made during the past 25 years. The total effort in this area has been vast, and any attempt to provide a comprehensive review would have been far beyond the scope of this paper. All that was possible was to highlight some of the key developments relating to chain codes, in particular, their generalization to straight and curved segments of varying length, as well as to some variations of these, including those based on the use of hierarchical approaches.

References

1) H. Freeman, "Boundary encoding and processing," in *Picture Processing and Psychopictorics*, ed. by B. Lipkin and A. Rosenfeld, Academic Press, New York, NY, 1970, 241-266.
2) A. Rosenfeld and A.C. Kak, *Digital Picture Processing,* v. 2, Academic Press, New York, 1982, p.196.
3) K. Dunkelberger and O. Mitchell, "Contour tracing for precision measurement, *Proc. IEEE Int'l. Conf. on Robotics and Automation,* St. Louis, MO, 1985, 22-7.
4) F.Y. Shih and W-T Wong, "A new single-pass algorithm for extracting the mid-crack codes of multiple regions," *Jour Visual Commun. and Image Repr., 3*, 3, Sept. 1992, 217-24.
5) C. Cherry. *On Human Communication*, John Wiley & Sons, New York 1957, pp. 46-7.
6) H. Freeman, "On the Encoding of Arbitrary Geometric Configurations", *IRE Trans., EC-10*, (2), 260-268, June 1961.
7) H. Freeman, "Computer Processing of Line-Drawing Images," *Computing Surveys, 6*, (1), March 1974, 57-97.
8) J. Koplowitz,, "On the performance of chain codes for quantization of line drawings," IEEE *Trans. Pattern Analysis and Machine Intell., Vol. PAMI-3*, (2), 1981, 180-185.

9) H. Freeman, "Application of the generalized chain coding scheme to map data processing," *Proc. IEEE Computer Soc. Conf. Pattern Recog, and Image Proc.*, Chicago, IL, 1978, 220-6.
10) H. Freeman and J. Saghri, "Analysis of the Precision of Generalized Chain Codes for the Representation of Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intell., Vol. PAMI-3*, (5), Sept. 1981, 533-539
11) D.L. Neuhoff and K.G. Castor, "A rate and distortion analysis of chain codes for line drawings," *IEEE Trans. Information Theory, IT-31*, (1),. 1985, 53-67.
12) C..Y Choo and H. Freeman, "Computation and features of 2-d polycurve-encoded boundaries," *Proc. IEEE Int'l. Conf. on Systems , Man, and Cybernetics,* Cambridge, MA, 1989.
13) L. O'Gorman, "Primitives chain code," in *Progress in Computer Vision and Image Processing*, ed. by A. Rosenfeld, and L. Shapiro, Academic Press, San Diego, 1992, 167-183.
14) H. Samet and R.E. Webber, "On encoding boundaries with quadtrees," *IEEE Trans. Pattern Analysis and Machine Intell., Vol. PAMI-6*, (3),. 1984, 365-369
15) W.G. Kropatch, "Curve representations in multiple resolutions', *Pattern Recog. Letters, 6,* (3), 1987, 179-184.
16) P. Meer, C.A. Sher, and A. Rosenfeld, "The chain pyramid: hierarchical contour processing," *IEEE Trans. Pattern Analysis and Machine Intell., Vol. PAMI-12*, (4), 1990, 363-376.
17) H. Freeman and J. Glass, "On the quantization of line-drawing data," *IEEE Trans. System Science and Cybernetics, vol. SSC-5*, (1), January 1969, 70-79.
18) H. Freeman, "Shape Description via the Use of Critical Points," *Pattern Recognition 10*, (3), 1978, 159-66.
19) C-H.Teh and R.T. Chin, "On the detection of dominant points on digital curves," *IEEE Trans. Pattern Analysis and Machine Intell., Vol. PAMI-11*, (8),. 1989, 859-872.
20) P. Zhu and P.M. Chirlian, "On critical point detection of digital shapes," *IEEE Trans. Pattern Analysis and Machine Intell., Vol. PAMI-17*, (8), Sept. 1995 737-748.